AD ASTRA...

Atari 1450XLD

16K → 48K upgrade

THE JOURNAL OF
THE ATARI MICROCOMPUTER NET
AMATEUR RADIO OPERATOR USERS' GROUP

AD ASTRA...

THE JOURNAL OF THE ATARI MICROCOMPUTER NETWORK

THE ATARI MICROCOMPUTER NET USERS' GROUP
NET COORDINATOR,
Jack McKirgan II, WD8BNG
4749 S.R. 207 N.E.
Washington C.H., Ohio 43160

JULY-AUGUST 1983

AD ASTRA... VOL. 2, # 2

The ATARI Microcomputer Net is a non-profit organization of amateur radio operators, short-wave listeners and ATARI Computer Enthusiasts who share a common interest- exchanging information on applications, programming and operation of the ATARI Microcomputer System. With these goals in mind, all persons are invited to join the net for the purpose of personal enlightenment and fraternalism. Amateur radio operators and short-wave listeners are especially encouraged to directly participate in the weekly on-the-air meetings.

"Ad Astra..." is the official journal of the ATARI Microcomputer Network and is made available to all registered members of the Net. "Ad Astra..." is an optional entity of the Net and there is no obligation to receive the journal. Members who wish to receive "Ad Astra..." are asked to help offset the cost of printing and postage by sending an annual donation of $10.00 to Net headquarters at the address shown at the top of this page.

"Ad Astra..." is published six times per year by the ATARI Microcomputer Network and all articles contained herein are submitted and published with no promise, implied or expressed, of monetary compensation. Neither the Net, it's officials, members or authors may be held responsible for the validity or accuracy of the articles contained within the pages of this journal.

EDITORIAL . . .

Dear Members,

We had a fantastic time at the Dayton Hamvention! Fortunately, we had the booth located inside the arena as the weather outside was attrocious! It rained for three solid days and because of that the fleamarket was full of great bargains. Just to let you know what you missed out on if you didn't attend or if you weren't observant: ATARI 810 drives for $385, Microtek 32K boards for $27, NEC 200ns 4164s- 8 for $30, DE-9S or DE-9E connectors (w/slimline hoods) $3... and on and on and on!

One thing was quite evident... we didn't have enough room! At one point on the Saturday of the event, we had folks 15-deep trying to get to the booth and the overflow was disrupting participation in a couple of adjacent booths. We will need to change our strategy for the next Dayton Hamvention! I want to issue a challenge!! **Who among our members would be willing to give a lecture on specific ATARI Computer System topics!** It would be held in a special meeting room right in the Hamvention arena! We could have two or three consecutive speakers on subjects as diverse as communications interfacing, computer assisted design, slow scan TV, RTTY/ASCII/AMTOR, analog-to-digital conversion, enhancing system design etc., etc., etc. If you would like to speak to your fellow net members in an informal meeting and share your experiences, this would be a great opportunity for you. Please contact me as soon as possible so that we can make plans with the Dayton Hamvention Committee and reserve a room for next year. Another thing that we may do differently is rather than set up a booth in the Hamvention arena, we may reserve a suite at one of the motels for a full gathering of net members. We could make this our "annual meeting", so to speak. Let me know what you think about this idea!

One advantage of having a booth at the Hamvention is that sooner or later, 99% of all attendees do drop by to see what's going on. We signed up about 50 new members who didn't even know that we existed! On top of that, we had registration forms for handouts at the event... we took 1000 of them and only brought home about 200!!! I'm writing this column only a week after the Hamvention and already I've started to receive some of those forms back! Our membership has steadily increased and it looks like the trend will continue! Of course, you are all responsible for this and I want to thank you all for helping to spread the word!

I recently received a newsletter from an ATARI users' group that contained a scathing condemnation of ATARI's attitude toward users' groups, hardware hackers, and individual users! There was a dissertation on the fact that the Commodore 64 has emulators that allow the use of other software in it's machine etc. and there was an open letter to ATARI condemming them as a greedy bunch who sent production facilities to Taiwan (which was incorrectly spelled in the aricle) and Hong Kong. The signature

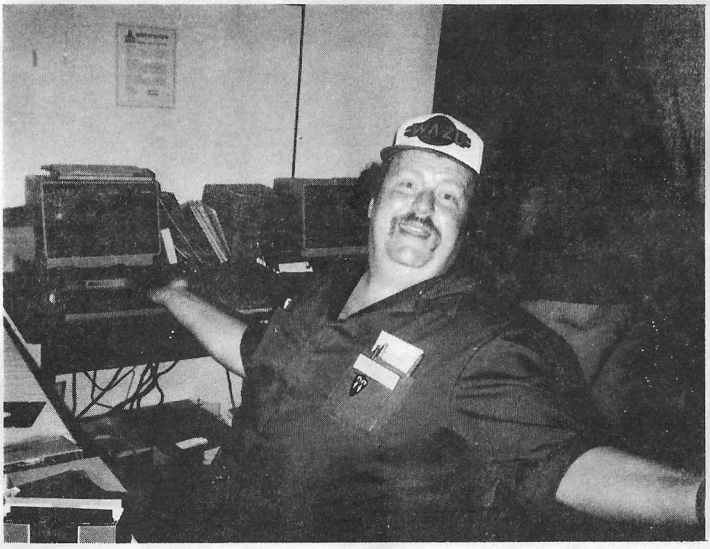
was followed by the title "¥ AMERICAN ¥". It was followed by a weak rebuttal by a person with the title "Another American"! While I don't pretend to know if these people have had experience with microcomputer manufacturers other than ATARI, I can tell you that I have! Let's look at facts rather than let emotions or patriotism taint our thoughts. Of course ATARI is greedy! I was always taught that the reason for being in business was to make money! The fact is that Commodore and Apple have manufacturing plants in the far east and can make their products rather cheaply. Even Radio Shack has computer manufacturing facilities in Mexico and a large part of TI's labor is Mexican rather than American. This is only natural when competition becomes so intense that you have to start finding ways to lower your costs. The fact also is that even if ATARI had kept manufacturing in Sunnyvale, new automated lines would have replaced many jobs anyway! What's so special about the Commodore 64 anyway? Commodore doesn't supply the emulators for making it think it is a TRS-80, ATARI 800 or Apple II! In fact it is third party material! I doubt that any of these persons have seen one of these emulators in action! The fact is that they just don't work unless the program is text-based only! Another fact is that many of these emulators are rip-offs and will probably never see the light of day! One of them was on display at the Las Vegas show last fall and it was showing all kinds of nifty Apple software in action! One enterprising fellow sneaked behind the booth and peeked under the table and 'lo-and-behold! There was a "PINAPPLE" (Apple II clone) motherboard mounted tightly under the table! Perhaps that company will produce a workable product... or perhaps they were high-tech rip-off artists looking for "investors"!? Have any of you ever delt with Commodore? I have! How about Tandy? I have! Or maybe TI? I have! The fact is that as disenchanted as some people may be with ATARI, they are the most receptive and helpful bunch that has ever populated the microcomputer market! A very close second goes to Apple, who, because they started earlier with their public and third-party support operations, have the lion's share of that support. The fact is that Commodore is surpassed in lack of total support or third party encouragement only by TI! At least ATARI, Apple, and Radio Shack are not embroiled in open warfare to the extent that they are robbing their customers when it comes to upgrades! Yes, I owned a Commodore 64! I can tell you that it can't hold a candle to an ATARI 800/1200XL or an upgraded 400! This second generation "friendly computer" is still using old PET 2.0 BASIC (and Commodore has announced that they will not offer an upgrade!). If the president of the ATARI users' group whose article promted me to write this editorial, had done his homework, he would know that ATARI has an upgraded ATARI BASIC coming that will be available in June. On top of that, there will be the ATARI Microsoft BASIC and shortly afterward, LOGO! Commodore's answer to questions about their BASIC is "we don't feel that persons using the Commodore 64 will be involved in BASIC program development." (March 1983 "PERSONAL COMPUTING"). INCREDIBLE!! As it is, if

you want direct sound and graphics statements with the Commodore 64, you must buy (yet another) "expansion" cartridge! (Thus releiving yourself of a large sum of cash and an addition 8K of RAM area!) Instead of printing a string at a specific location or plotting graphics or creating certain sounds with your ATARI computer, pull out a memory map and try POKEing them all in. You'll soon see just how "friendly" the Commodore 64 really is! Oh, yes! Also try getting some information from them on the so-called "user port" or serial port protocall... I tried for 7 weeks... at my expense on the phone and the only response that I ever got was "buy our printer"!

What this all boils down to is that some of us tend to confine ourselves to a small corner of our own world. While in that corner we tend to do one of two things: Either complain about the state that we are in because we haven't pulled our heads out of the sand long enough to see whats really going on, or be extremely defensive about the product or service that we have committed ourselves to. I personally prefer to see what the other guy is doing... maybe I can benefit from his mistakes or successes. Neither should I ignore my own mistakes of the past. I think that is what ATARI is doing too!

73,
Jack

Jack, WD8BNG



Rick Walsh, WD8HSJ
"Happiest" member
at the Dayton
Hamvention!

EDITORIAL II...

This is a first! Actually it is a second.... editorial, that is! I have just returned from the Summer Consumer Electronics Show in Chicago...as the guest of ATARI!!! Mark Cator, assistant director of the ATARI Users' Group Support Team, called me to tell me that a ticket was waiting for me at Port Columbus and that I should meet him in Chicago! Short notice it was, but I grabbed the opportunity! There were at least ten representatives of large ATARI Computer Enthusiasts groups on hand and we managed to exchange a few pleasantries before the BIG meeting at which ATARI explained what they were doing and also gave hints at what was planned for the future.

I was pleased to see several things brought out at the meeting.... perhaps I should itemize:

1. ATARI has not been sitting on their laurels- they have smart people doing smart things with a huge backing in the R&D departments. The four new computers are indeed an extension of logical thinking within the company.

2. ATARI is aggressively going after the home video and computer market with great talent. They have proclaimed that there is no resource that they won't tap... including providing software for other computer systems.

3. ATARI is continuing present user support and is starting new programs- good for us all.

4. ATARI is listening to the end-users and the dealers- they want to know what you want!

That was a report of the facts that were presented. Now for the real editorializing....

Most of the user group representatives that I met were as awestruck with the proceedings as I was... I was not used to being treated like royalty and in fact, while I enjoyed it, I'm not sure that it was necessary. Sure, the industry big-wigs and distributor's reps were used to it and it was perhaps proper as they were being courted for sales. In the case of the users' group reps, certainly merely being there as a guest was honor enough!

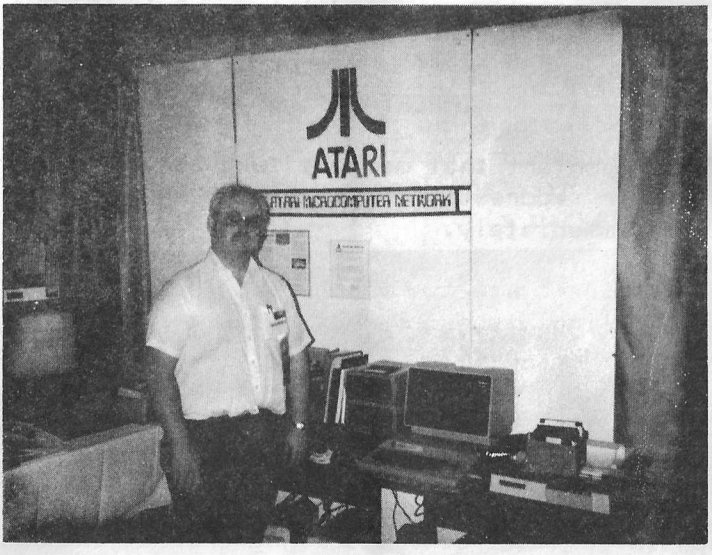
I took this meeting as an opportunity to be a reporter of the events. Unfortunately, a few of the reps from large users' groups used the occasion and even the hospitality of ATARI as a forum for expressing personal opinion and perform a general feeding of the ego. I really have to feel sorry for Earl Rice and Mark Cator, for it was apparent that they are often

between the proverbial rock and a hard place! They do their best to help all groups with the resources that they have, but to some self-indulgent pundits of the "Super Groups", the best is never enough. I receive many newsletters each month from many of these groups and many of them contain some of the most cynical "amusment articles" that I have ever read. They are full of pseudo-parables, supposition and display just plain ignorance about electronics, the machine that they use and real-world marketing. My advise to these persons is to try to get similar support from Commodore, TI, or Radio Shack! Some if these groups don't even know about how to maintain status as an official ATARI Computer Enthusiast (ACE) group! But they are quick to point out how certain other groups have not continued as an "official" group! It seems to me that they should realize that ATARI and Warner **do not** revolve around them!

I may soon be known as the Copernicus of the ACE groups and be chastized for speaking the truth, but at least I can take comfort in the fact that I was one of the truly faithful and did not require ATARI to perform **DAILY MIRACLES.....** once every four years is quite enough!!!!

DE Jack, WD8BNG

A personal **THANKS** to Earl, Mark and ATARI!



A slightly disheveled WD8BNG at the ATARI Micro-Net's booth during the Dayton Hamvention

MEMBER SERVICES

DISKETTES W/SLEEVES

We are now able to obtain diskettes with sleeves at a low price. Previously, the sleeves were an additional cost due to the bulk-style packaging of the disks. These disks could be one of several brands as we receive only what is available at the moment from the supplier. These brands have been Wabash, Memorex, Scotch and Verbatim in the past. Cost from Net HQ is $2.00 per diskette. Shipping is included in orders for 5 diskettes or more. If the order is for less than 5 diskettes, please enclose an extra $1.00 to cover the postage. The profit ($.40 less postage) goes into making "Ad Astra..." bigger and better!

DISKETTE STORAGE BOXES

We have on hand a small number of plain white boxes of the type that diskettes are usually purchased in. These boxes are available for $.50 each. Send an 8 X 10" envelope with enough postage for your boxes. Each box weighs approx. 1 oz. We will investigate the possibility of printing the "Ad Astra..." logo on the boxes at a later date!

IMPORTANT!

It is VERY important that members who have moved or changed their address to contact Net HQ with the new information immediately.

Also, if you feel that the "subscription" information on your mailing label is not correct, please send a photocopy of your check or a copy of your confirmation letter (the letter that was sent to you when you registered with the net.)

I try very hard to keep all information current and I have 2 separate data bases for all members. Of course, it IS possible that I goofed somewhere along the line! Let me know if you think I did!

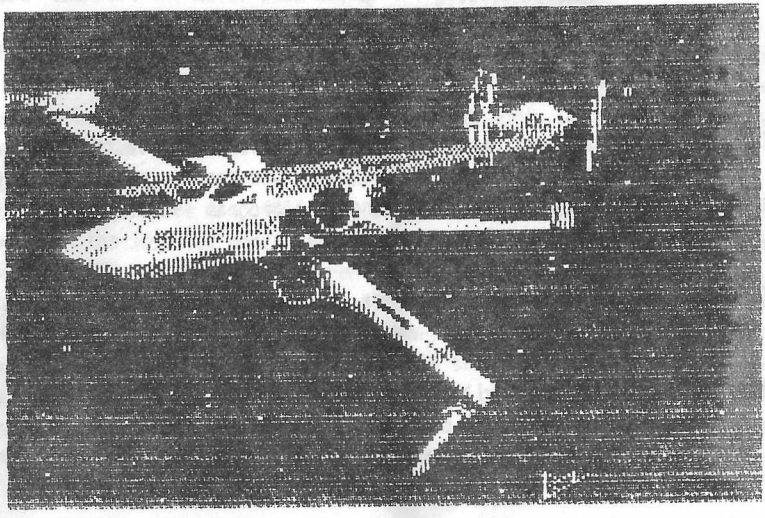
THANKS!

Jack, WD8BNG

NET ORGANIZATION

Regional calling frequency: 7.235 Mhz (Call station or CQ ATARI)
National Net: 14.325 Mhz. at 1600Z, Sundays, NC/WD8BNG
Midwest Regional Net: 7.235 Mhz. at 1830Z, Sundays, NC/WD8BNG
Southeast Regional Net: 7.235 Mhz. at 1800Z, Sundays, NC/KD4DB
Southwest Regional Net: 7.230 Mhz. at 1800Z, Sundays, NC/KC5FW
Pacific NW Regional Net: 7.230 Mhz. at 1800Z, Sundays, NC/KC7DG
East Coast Regional Net: 3.960 Mhz. at 8 pm EST, Wednesdays, NC/N2CZW
West Coast Regional Net: 7.235 Mhz. at 11 am PST, Sundays, NC/WA6TUB
International Net: 21.400 Mhz. at 2330Z, Alternate Thursdays, NC/WD8BNG
Dayton, Ohio Local Net: Open channel daily on 146.445 Mhz., Simplex
Chicago, IL Local Net: Open channel daily on 147.570 Mhz., Simplex
Central Kentucky Local Net: 145.85 (TX 600Khz down) repeater, 8 pm EST, Wednesdays, NC/WD4HPL

Additional nets will be formed as regional/local net control stations volunteer their time. If you would like to start a regional/local net in your area, contact WD8BNG for a Net Coordinator's packet.



48/64K UPGRADE FOR THE ATARI 400

by Claus Buchholz

EDITOR'S NOTE: This article originally appeared in the "MACE NEWSLETTER", September 1982 issue and also appeared in Volume 1, # 4 of "Ad Astra..." shortly thereafter. At that time the net only had 200 members and we have more than tripled our size since then. With the price of '400s plummeting to less than $70 during the model changeover, this article can be of great value as the memory chips can now be purchased for less than $40 per set! Have fun and be careful! DE Jack, WD8BNG

Nonetheless, we know that among our members there are a few incorrigible hackers who think that hardwired spaghetti improves the machine's asthetic value, as well as some who can't resist a bargain. Although we don't want to encourage you, we would rather have you down in the basement ripping your computer apart than out on the streets where you might do some real harm. So in the interest of public safety, we publish the following article. We suggest that you have a hardware manual handy as well, to refer to the schematics and block diagrams. After all, you've got almost $250 invested in your computer!]

None of us needs to be reminded of the awsome power of the ATARI personal computers. What many fail to realize is that, except for the full-stroke keyboard and greater configurability of the '800, the ATARI 400 shares all of the power of her big sister. The high performance/price ratio of the '400 makes it a very attractive computer.

The 16K RAM supplied (8K in earlier models), however, is simply inadequate for many users' needs. ATARI designed the '400 to address 32K but they don't sell 32K boards. Other manufacturers sell 32K and 48K boards, but their added cost severely decreases the performance/price ratio that distinguishes the '400 from other computers.

I have designed and implemented a 48K upgrade for the '400 that you can add for about $70 and a few hours work. With 48K, you can run nearly every program written for the ATARI computers, including that program you've not finished writing because, "It won't fit!"

The modification is based on the idea of replacing the existing 16K-bit (or 8K) RAM chips with the newer 64K-bit devices. These dynamic RAMs are operationally compatable with the 16K chips. Note the two major differences: The 64K RAMs have an additional multiplexed address pin to access the larger memory. Also, they need only a single 5V power supply as opposed to the 5V, 12V and -5V

supplies which the 16K RAMS use (see Figure 1 for a pinout comparison).

Some circuitry must also be added to allow the '400 to address 48K. Note that the new RAM chips can hold 64K of memory, but the ATARI only addresses 48K. If you can't bear to waste the extra 16K, see the suggestions later in the article.

The parts listed in the Parts List are available from many mail order houses who advertise in the back of most computer magazines. You will also need a fine-tipped soldering iron, an ohmmeter, small pliers, screwdrivers, solder, fine wire, and a clean and static-free place to work. You should have a little experience in working with electronics. If you don't find a friend who does and could help you.

The first step is to open your '400. Disconnect all cables. Turn the '400 over and remove the four screws in the underside of the plastic case. While holding the case together, turn it over again. Open the cartridge door and remove any cartridge, leaving the door open. Lift the rear of the top-half of the case over the door. To remove the case top from the keyboard, press on the bottom of the keyboard on either side until it bends, and slide the keyboard away from you. The case top should now be free. Now remove the keyboard by pulling straight up on the flexible connector under the right side of the keyboard.

The circuit board on the right is the power supply. The computer is inside the metal case. Remove the two screws that fasten the left side of the power supply board to the right side of the metal case. Gently, but firmly pull up the left-front side of the power supply to disconnect it from the main board on the bottom. Be careful of the plastic interlock switch plunger when moving the power supply board. Now remove the speaker connector from the left-front of the main board, and lift the metal case out of the plastic bottom.

Turn the metal case over and remove all the screws in the bottom plate. Now pull the main circuit board up and out of the metal case, taking care not to flex the board. You may have to gently pry the edges to loosen the board from the metal case.

You will now see the '400 in it's full splendor. Lay the main circuit board down so the joystick ports face you. The smaller boards sticking up are memory board and CPU board. The one nearer you is the memory board. Unplug each, again being careful not to flex the circuit boards. You may also remove the beige plastic piece on the main board by bending it's prongs underneath the board.

Look at the CPU board. It has three large chips. The middle one is the CTIA or GTIA. If you want to replace your CTIA with a GTIA, now is the time to do it. The CPU board is not altered in this memory upgrade, so put it away.

Look at the memory board. The eight chips along the top are the RAM chips. The other four chips are the addressing circuitry. The edge pin connectors at the bottom are labeled as in Figure 2. If you have an 8K '400, you must alter the memory board before proceeding with the upgrade. Instructions for this modification appear at the end of the article.

The first step in the 48K modification is to eliminate the 12V and -5V sources on the board and move the 5V source to where the 12V used to be. As shown in Figure 3, cut the trace going from pin "X" of the board's edge connector to the capacitor C521. Also cut the trace going from edge pin "Y" to C523. Cut the traces cleanly and completely. Be careful not to slip and damage adjacent traces.

Now remove the capacitors C521 and C523. The trace coming from pin "W" carries 5V. Using a short piece of wire, make a solder bridge between this trace and the old 12V trace, at the point where C523 used to be (see Figure 3). Next, remove the eight capacitors C503, C505, C507, C509, C511, C513, C515 and C517, which are usually in a row along the top of the board.

We now have 5V going to pins 8 and 9 of the RAM chips, and no connection to pin 1. Remove the eight RAM chips and insert the 64K RAMS in their place, properly orienting the notched ends. With an ohmmeter, make sure there is no connection between edge pin "Y" and pin 8 of the chips, nor should there be any connection between any two of the edge pins "W", "X" and "Y".

If all has gone well, the board should be functioning exactly like a 16K memory board, since the addressing circuitry has not been altered. Now may be a good time to test the board (particularly the new RAM chips). If you wish, reassemble the entire computer and check to see if it works properly as a 16K '400. If it doesn't work, recheck all connections and disconnections made so far.

FRE(0) = 13326

Now take the 5V supply off pin 9 of the RAM chips. To do this, cut the rightmost wide trace on the chip-side of the board (see Figure 4).

Pick up the 74LS158 chip, which is the same as the chips Z503 and Z504 on the memory board. With needlenose pliers, carefully bend up all pins except 1, 8, 15, and 16 (see Figure 5). The remaining four pins are to be soldered to the chip Z503. Remove the chip at Z503 from it's socket and place the 74LS158 on top so that the four pins listed above touch the same four pins on the lower chip (as in Figure 5). Carefully, solder each of the four pairs together, being careful not to get too much solder on the end of each pin.

Now solder a 4" length of wire to each of the pins 2,3 and 4 of the top

PARTS LIST

QTY	ITEM
8	4164 200 nanosecond dynamic RAM
1	74LS158 quad 2 to 1 multiplexer
1	74LS02 quad NOR gate
2	680 ohm 1/2 watt resistor
1	14-pin DIP soldertail socket

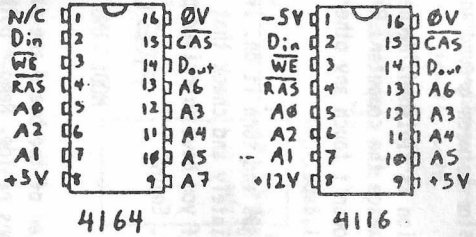


Fig. 1 - Pinout comparison of 64K- and 16K-bit RAMs

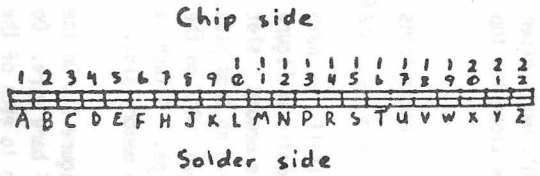


Fig. 2 - Connector identification for memory board, seen from below

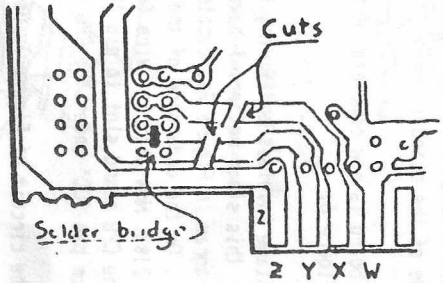


Fig. 3 - Lower left corner of solder side of memory board

chip. Reinsert the chip pair at Z503. Solder the wire from pin 2 into the hole attached to edge pin "M", and the wire from pin 3 to edge pin "U". Next solder the wire from pin 4 to a hole in the former 5V bus, the wide trace along the top of the chip side of the board.

The memory board is now complete. With an ohmmeter, check all connections diagrammed in Figure 6.

The final stage involves modifiying the main (mother) board itself. To help you visualize this stage better, I have included a partial schematic in Figure 7, and a pin diagram in Figure 7a. Locate chip Z103 forward of the memory slot (see Figure 7a). On the underside of the board, cut the traces leading from pins 1 and 2 of Z103. Now attach a wire from pin 24 (across from pin "BB") on the underside of the CPU board slot to pin "U" under the memory slot. Attach a second wire from pin "CC" under the CPU slot to pin "M" under the memory slot.

Now wire the circuit of Figure 7, using the pin diagram of Figure 7a. On the 14-pin socket, solder pins 3 and 4 together with a short piece of bare wire. Do the same with pins 2 and 13. Next solder an 8" length of wire to each of the pins 1, 5, 6, 7, 11, 12 and 14. With these wires, make the six connections to the underside of the cartridge slot as diagrammed. The seventh wire from pin 1 goes to pin 18 on the underside of the memory slot.

Plug the 74LZ02 into the socket and bend the wires around some notches on the edge of the main board, between the crystal and cartridge slot. Finally, solder one of the 680 Ω resistors between pin "A" under the cartridge slot and the nearest ground connection. Be especially careful that excess solder does not form "bridges", making electrical connection where none should exist. Put the second 680 Ω resistor between ground and pin 14 under the cartridge slot.

The modification is finished. Recheck all connections, as an improper connection may damage the computer. Reassemble the computer, being careful that the 74LS02 chip doesn't touch any other circuitry. It's a good idea to wrap the chip in electrical tape.

Plug in the '400 and turn it on. If the blue screen doesn't come up quickly, turn it off immediately and check that your work, including reassembly, has been done correctly. If you have exercised proper care, you should now have 48K of RAM for your '400. Enjoy!

FRE(0) = 37902

MODIFYING AN 8K BOARD

Near the center of the board are six pair of holes marked A through F in which two resistors reside. Remove both resistors. If one of them is at C, leave it there. Otherwise, solder one of the removed resistors at C. Now solder a wire

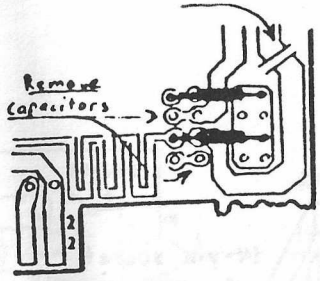


Fig. 4 - Lower right corner of chip side of memory board

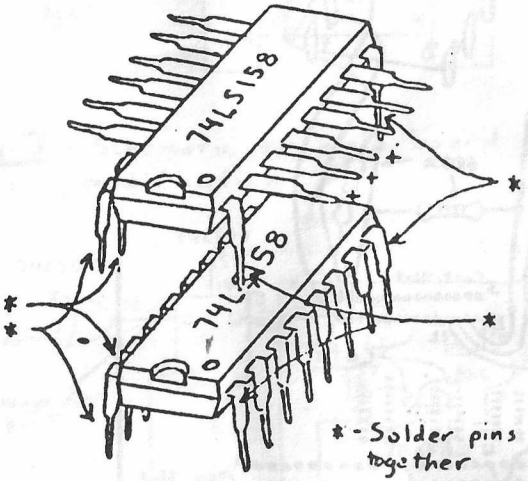


Fig. 5 - Piggyback arrangement

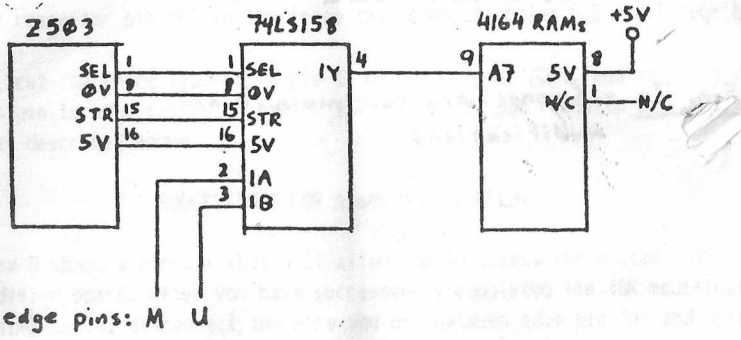


Fig. 6 - Schematic for memory board modification

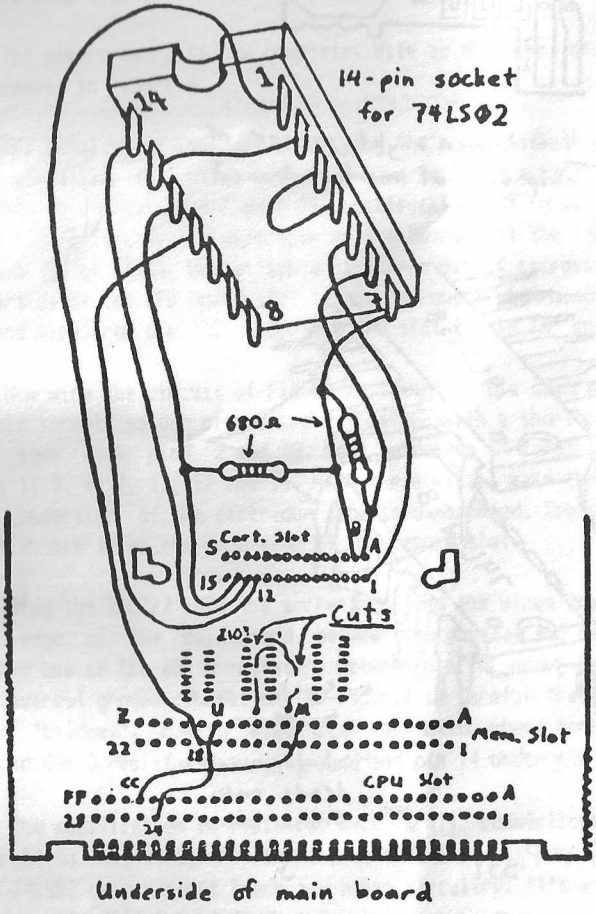


Fig. 7a - Connections for main board modifications

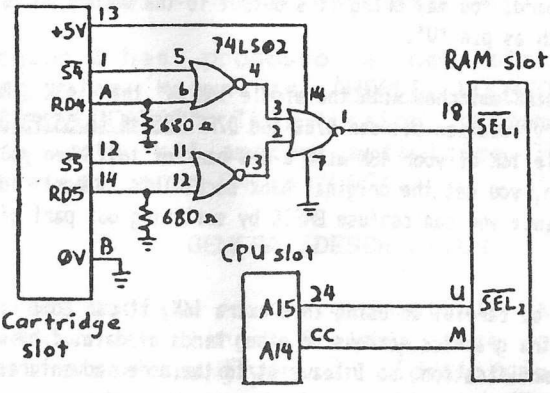


Fig. 7 - Schematic for main board modification

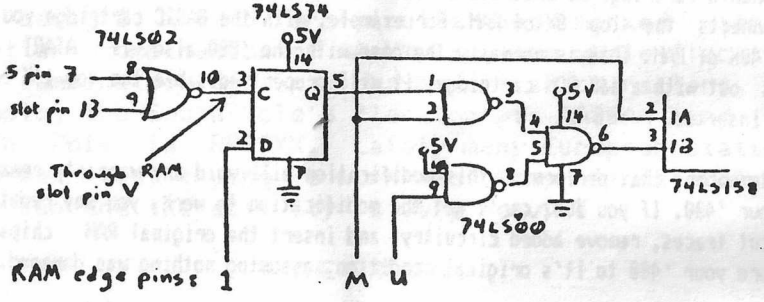


Fig. 8 - Schematic for 64K modification

from edge connector pin "H" to the trace that connects holes D,E and F together.

Next, cut the trace leading to pin 13 of the chip at Z501, and solder a wire from this pin to edge connector pin "U". The board is now ready to be modified for 48K as described above.

SUGGESTIONS FOR A 64K MODIFICATION

Figure 8 shows a circuit that will allow you to access the unused 16K on your modified board. After you have successfully completed the 48K modification as described above, disconnect the wire you put between edge pin "U" and pin 3 of the 74LS158. Wire the circuit of Figure 8 in it's place.

Two more chips are needed for this circuit, a 74LS00 quad NAND gate, and a 74LS74 dual flip-flop. They may be wired to the memory board using sockets as you did with the 74LS02. The NOR gate on the left is from the 74LS02 chip you wired to the main board. You may bring it's output to the memory board through an unused edge pin such as pin "U".

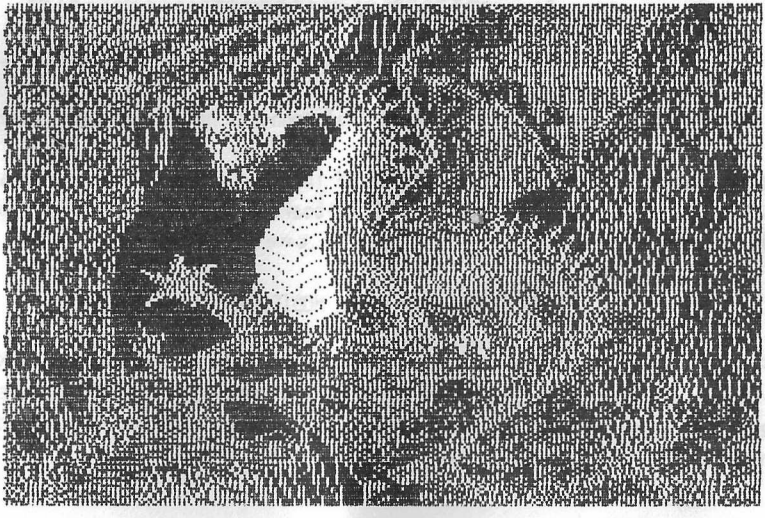
The extra 16K is bank switched with the middle 16K of the 48K RAM. By writing a 1 to a memory location between D700 and D7FF (55040 to 55295 decimal), you replace the middle 16K of your 48K with a new bank of 16K. When you write a 0 to the same location, you get the original bank back. This is best done in machine language, since you can confuse BASIC by switching out part of a BASIC program.

Although you must be careful in using this extra 16K, it can come in very handy for storing extra graphics screens or other kinds of data. I have not yet implemented this 64K modification, so I leave it to the more adventuresome of you to build, test and use.

FINAL NOTES

When a cartridge is inserted into the '400, the addressing circuitry disconnects the top 8K of RAM. For example, with the BASIC cartridge you only have 40K of RAM. This is normally the case with the '800 also. If ATARI ever comes out with a 16K ROM cartridge, it will properly disable the top 16K of RAM when inserted.

Remember, that performing this modification will void any warranty remaining on your '400. If you just can't get the modification to work, you may repair all the cut traces, remove added circuitry, and insert the original RAM chips to restore your '400 to it's original condition, assuming nothing was damaged.



FINDING YOUR OWN "LOCATOR"

by Zvonimir Makovek, YU3HI

IARU Region I has proposed a new standard location plotting plan known as "WORLD LOCATOR SYSTEM" or "UNIVERSAL LOCATOR". This locator system is intended for use with all amateur activities, HF and VHF/UHF. The abbreviation on CW is "LOC".

GENERAL DESCRIPTION

The earth's surface is divided into 18 X 18 segments (324) known as large fields, each one is 20 X 10 degrees and each is given an identifying mark of a 2-letter combination between AA and RR. Each of these large fields is divided into 10 X 10 (100) fields, each being 2 X 1 degrees and identified with a 2-number designation between 00 and 99. Each of these units is further divided into sub-fields of 24 x 24 units (576), each being 5 X 2.5 arc-minutes and marked with a 2-letter combination of AA-XX. So, the whole "locator" is a combination of six alpha-numerical characters. For example, the South Pole's "locator" is AA00AA and the North Pole is RR99XX. Lately many European stations have been busy trying to work as many "locators" as they can and the activity is very heavy.¥

THE PROGRAM

The following program will allow you to convert your standard geographical coordinates into your "locator". The program is written in standard ATARI BASIC. A word of warning: You MUST input your longitude with three (3) numbers in the DEGREES section (e.g 75 degrees = 075). To do otherwise will lead to an input error.

```
1 REM COORDINATE--> LOCATOR BY MAKI, YU3HI
10 DIM A$(7):? CHR$(125);CHR$(29);CHR$(29)
20 ?"COORDINATES--> LOCATOR": ?
30 ?"INPUT LONGITUDE":? "DDDMMSS      ";:INPUT  A$:  LO=
VAL(A$(1,3)) +VAL(A$(4,5))/60 + VAL(A$(6,7))/3600
31 ? "EAST/WEST     ";:INPUT A$
32 IF A$(1,1)="E" THEN 40
33 IF A$(1,1)="W" THEN LO=-LO:GOTO 40
34 GOTO 31
```

```
40 ?:? "INPUT LATITUDE": ? " DDMMSS        ";:INPUT  A$:
LA=VAL(A$(1,1)) + VAL(A$(3,4))/60 + VAL(A$(5,6))/3600
41 ?"NORTH/SOUTH ";: INPUT A$
42 IF A$(1,1)= "N" THEN 50
43 IF A$(1,1)= "S" THEN LA=-LA:GOTO 50
44 GOTO 41
50  LO=(LO+180)/20:  LA=  (LA+90)/10:  A=  INT(LO):  B=
INT(LA):  LO=(LO-A)  *  10: LA= (LA-B) * 10: C=INT(LO):
D=INT(LA)
60 A$(1,1)= CHR$(A+65): A$(2,2)=  CHR$(B+65):  A$(3,3)=
CHR$(C+48): A$(4,4)= CHR$(D+48)
70   A$(5,5)=CHR$(INT((LO-C)   *24)+65:   A$(6,6)=
CHR$(INT((LA-D) * 24)+65)
80 ?:? "LOCATOR  =  ";
90 FOR E=1 TO 6: ? CHR$(ASC(A$(E,E))+128);: NEXT E
```

* This looks like a candidate for a new contest! Worked All Locators! Ed.

ADDITIONAL COMMENTS

Although I don't have an expensive printer, I am able to use my old teleprinter machine. I have written a "TTY-handler" program in machine language for the ATARI, which can be booted from cassette and it sets all parameters (LOMEM, etc.) so you can use it with BASIC or any other language. It includes a "screen print" utility. Output is via one of the player-port pins and all the hardware needed is an AF transistor and a relay which keys the TTY machine. My TTY machine cannot print all of the ASCII characters, but this arrangement is better than nothing. A copy of my program "TTY-Handler" on cassette is available to members for an SASE with blank cassette + $1 U.S. to my address: Zvonimir Makovek, YU3HI, Box 1, YU-69240 Ljutomer, Jugoslavia
73, DE Maki, YU3HI

P.S. My "locator" is JN86CL, HI HI!

REPORT FROM CES
by Jack McKirgan II, WD8BNG

I was pleased to receive a call from Mark Cator, of ATARI's User's Group Support Team, informing me that I was to be a guest of ATARI at the Summer CES! Packing up my cares and woes I went to Chicago with and was pleased to find that ATARI was the **STAR** of the show! Among all of the booths shoveling out hoopla on new video disks, sound systems and various razzle-dazzle items were the computer manufacturers. Most of them were low-key and very business-like in their presentations.... some (and I'm talking BIG names) were almost as lonely as the Maytag repairman. One, Texas Instruments, didn't even show and all of the media seem to have picked up the phrase "**Texas Armadillo**" when speaking of that company. It seems as though TI's penchant for wanting to be the sole supplier of hardware and software for their home computer system has turned off a lot of vendors who were leaning toward support of that system last year. About two months ago TI announced that they would not be shipping the 99/4A **WITHOUT** the graphics ROM. This is the equivalent of leaving GTIA out of your ATARI system! This move forces the software vendor into selling their program to TI for exclusive distribution rights. Since TI will not license their Graphics ROM (GROM) to any other vendors, they must have the GROM on-board the cartridge (cartridges are one of the greatest ways of obtaining BIG profit margins in home computers). Spinnaker and several other vendors have announced that they want no part of this blackmail and have pulled out of their plans for supporting the TI machine.

Commodore and Radio Shack did not fare well either! Of course Radio Shack has their own distribution network which increases the overall profits of their systems. Commodore only had the previously-announced portable version of the 64 on hand and distributor reaction was very limp. The word going around McCormick Center was that even though Commodore has sold a bunch of low-end computers, they were losing money and the MOSTEK Divsion (semiconductor manufacturing) was not able to hold the computer division's head out of water. Third-party vendors were busy showing all of the new

boards and add-ons for the IBM PC and a few new ones for the Apple series. Media people were straining to get a glimpse of new equipment or an interview with someone who could supply more than cheesecake for the masses. Sanyo is going to be introducing an IBM PC compatable machine for less than $1000 and IBM is going to produce an anti-Apple machine! My my, how far can these computer wars go!!??

ATARI's NEW MACHINES

ATARI pulled off a coup détat by introducing four new machines... not **ALL NEW**, but new **AND SIGNIFICANT!!** With their present marketing plans ATARI will cover every price range from $199 to $499 in $100 jumps. And above that is a new model that "has it all", including a disk drive! We shall now describe the units:

ATARI 600XL

The new ATARI 600XL has the new, standard XL operating system with it's built-in system diagnostics and a beautiful-feeling **full-sized, full-stroke keyboard.** It also has **ATARI BASIC REV. "B" built-in!!** I was unable to find out if there were any enhancements in **REV. "B" BASIC** other than having some of the bugs in the old BASIC removed. All of the other enhancements of the 1200XL have been included, with the international character-set and music synthesizer as standard equipment. Standard RAM configuration is 16K, expandable to 64K via a plug-in board. The language is switched out whenever a ROM cartridge is plugged into the single top-mounted cartridge slot. Some of the **best news** is that there is a **CPU/OS bus on the back of the computer!!!** Great news for expansion buffs! The rest of the unit is pretty straightforward and it resembles a 1200XL with about half of the depth of the latter unit. Also, the special function keys are in a vertical row on the right side of the keyboard, much like the present 400/800 models. Video is limited to modulated video only a la the ATARI 400. Definite list price is $199.

ATARI 800XL

The ATARI 800XL is similar in layout and function to the 600XL except that the case is a little deeper and

it comes with 64K RAM as standard memory configuration. It also includes a monitor output as well as the built-in video modulator. I was told that the monitor output levels have been boosted to be able to drive any composite monitor and I was told, but not able to confirm that RGB monitors may also be supported. List price: $299!

ATARI 1200XL

The current 1200XL will remain in the $399 slot for a while. I was not able to find out if it will now be packaged with built-in ATARI BASIC and if the CPU/OS buss will be added... as well as other improvemnts. There seems to be some confusion as to whether the 1200XL will or will not remain in ATARI's product line.

ATARI 1400XL

This beauty is the same physical size of the 1200XL and contains all of the above... with two SIGNIFICANT FEATURES... It also has a **BUILT-IN MODEM and a BUILT-IN VOICE SYNTHESIZER!!!** This proves to me that ATARI has gone to great pains to provide every possible user with the machine that suits them best! The special function keys are located on the top-row in the same manner as the 1200XL. I was unable to confirm that the modem is treated as an RS-232 device, so be sure that your favorite communications program will support it! List price on this honey is $499!

ATARI 1450XLD

Basically the same unit as the 1400XL, the significant addition to this unit is **a built-in DOUBLE-SIDED, double density drive!!!** There is also room for a second drive unit in the sleek, low profile case, or you can use the empty space for safely stashing your diskettes that are to be used during a session! I do not know for sure, but I was told that this is a parallel-fed drive which will greatly speed-up I/O! This unit shows special attention to it's raised rear section in that it is specially reinforced and shielded for placement of a monitor above the disk drive area. List price on this **SUPER PACKAGE is ONLY $799!!!** Don't ask me how they are going to do it!

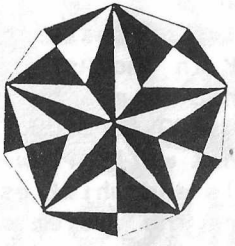
Now, you may ask, "What kind of support can I expect?"... FEAR NOT! ATARI has made it's intentions clear... almost UNLIMITED support and expansion to the system will be offered! To begin with: a new Double Density disk drive should be hitting the streets at any time. Price is expected to be much less than $400 (Possibly $300!). It will be supported by DOS 3.0 which should work right alongside DOS 2.0 on the 810 drive. You can also expect the long-awaited 835 Modem at any time. Also, look for yet a **NEW PRINTER... NOT** dot-matrix, but a true letter-quality machine that will retail for less than $400! As far as I could tell, the characters are formed from roller-drums with combinaions of segments producing the complete character. If this is indeed the case, then special characters could be formed with the right progamming or by changing the drums to obtain a special font! Add to all of this a true graphics tablet for $80 and a $50 light-pen that performs as well as any $450 professional unit that I have ever seen and you have a SUPER SYSTEM!!!

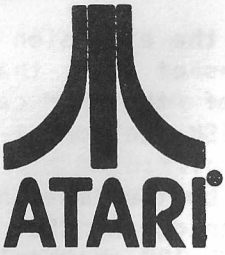
Perhaps the **Biggest** news is a full-fledged expansion module that will plug right into the CPU/OS buss on the new machines (and possibly on the undocumented buss of the present 400/800 systems!). Unlike the tripe that I have recently read in other ATARI users' group publications about this being yet another "special connector", it simply uses a standard edge-card that you can pick up anywhere. The new module will give you all kinds of I/O capabilities (I may be wrong, but I think I counted no less than 8 I/O ports!) plus buss-expansion of the system! What kind of buss expansion? How about a voice recognition card?! How about new OS/CPU cards? Yes, CP/M is supported! ATARI is **openly encouraging third-party support for this system... to the point that every CP/M card will have a catalog from "ADD-ON COMPUTER CORP." included! ADD-ON is a direct-marketing vendor of CP/M software and will be providing CP/M 2.2 software pre-configured for the ATARI system! One of the members of this organization, David Gangola, was a technician and designer at North Star Computer, Inc. and was chief designer of the CP/M card for the ATARI system. The card, by the way, includes 80-column capability as well as CP/M and the Z-80 CPU!!! The expansion unit will have 8 slots for additional cards (which is one more than the Apple][e!). It is unknown at this time what the total cost**

of the CP/M card and the expansion module will run, but I have heard of prices at less than $400 complete!!! Another advantage of the CP/M card is that it can be configured as a RAMDISK for normal OS operations.

I was most impressed by the "meeting of the minds" at the 1st Chicago Center during CES. It was apparent that ATARI has decided that the market is now expecting only the best from a computer system at the best prices! It was also clear that these gentelmen really do know their business and that they are giving a total commitment of resources to produce machines that the public wants and needs regardless of price catagory. They are definitely producing the highest quality units with total factory support to both the end-user as well as the distribution system. There is absolutly no rivalism at ATARI now that every section has merged into what they are calling "The NEW ATARI". The world is now at our fingertips and the program-base for the ATARI system just doubled in size due to the CP/M OS now being made available from the first-party! Now it's time for us all to await the first deliveries of the new standard in home computers. Deliveries of the new system should start in late September with CP/M available in October. I'm sure after all the bugs are worked out we will experience the difficulties that are documented in the old country-western song entitled: "Oh Lord, it's Hard to be Humble... When you're Perfect in so Many Ways"!! End of File. End of exclamations. Beginning of Domination.

DE Jack, WD8BNG





CLASSIFIEDS

WANTED: Old copies of computer magazines. Donations would be welcome because I cannot get specialized computer magazines in Yugoslavia. TNX. Zvonimir Makovek, YU3HI, Box 1, YU-69240 Ljutomer, Jugoslavia

I would be interested in trading programs from my library. Please send your list and I will reciprocate. Roger Bonnett, WB9NOE, 1300 Ann St., Harrisonville, MO 64701

I would like to contact other members of the net who are experienced in AMTOR communications. Bruce Crawford, WA3WUL, Five Boradbent Rd., Wilmington, DE 19810

I would like to contact other machine language programmers with the intention of combining efforts to produce a comprehensive RTTY/ASCII/CW/SSTV package. DE John Day, KA4CUB, 70 Bluebird Blvd., Indian Harbor, FL 32937

I am interested in trading programs from my library. Please send me your list and I will do the same. Or call after 6 pm EST. Jim Burkhard, KA2KGT, 7 Fairway Place, Boonton, NJ 07005, (201) 335-3278

AT RANDOM

From Bruce, WA3WUL: I find that the "HASH TABLE" as converted by KA4ATK in "Ad Astra..." works very well. But on my 8K '400 I had to change the number 11691 to 3891 in line 40 and 11700 to 3900 in line 70 and 450 to 150 in line 120. This will allow me to have 721 bytes free to add some bells and whistles. I can enter about 350 callsigns before the system crashes (I never have any more contacts that that anyway!).

CONNECTORS!

Hunting those elusive DE-9S connectors and slim hoods that you can plug into your ATARI's front panel? Contact CONNECTOR SPECIALTIES CO., INC., 416 E. 30th Street, Baltimore, MD 21218, (301) 467-1350. Chuck Burke can get you just about anything that you will need!

NEW ATARI MAGAZINE!

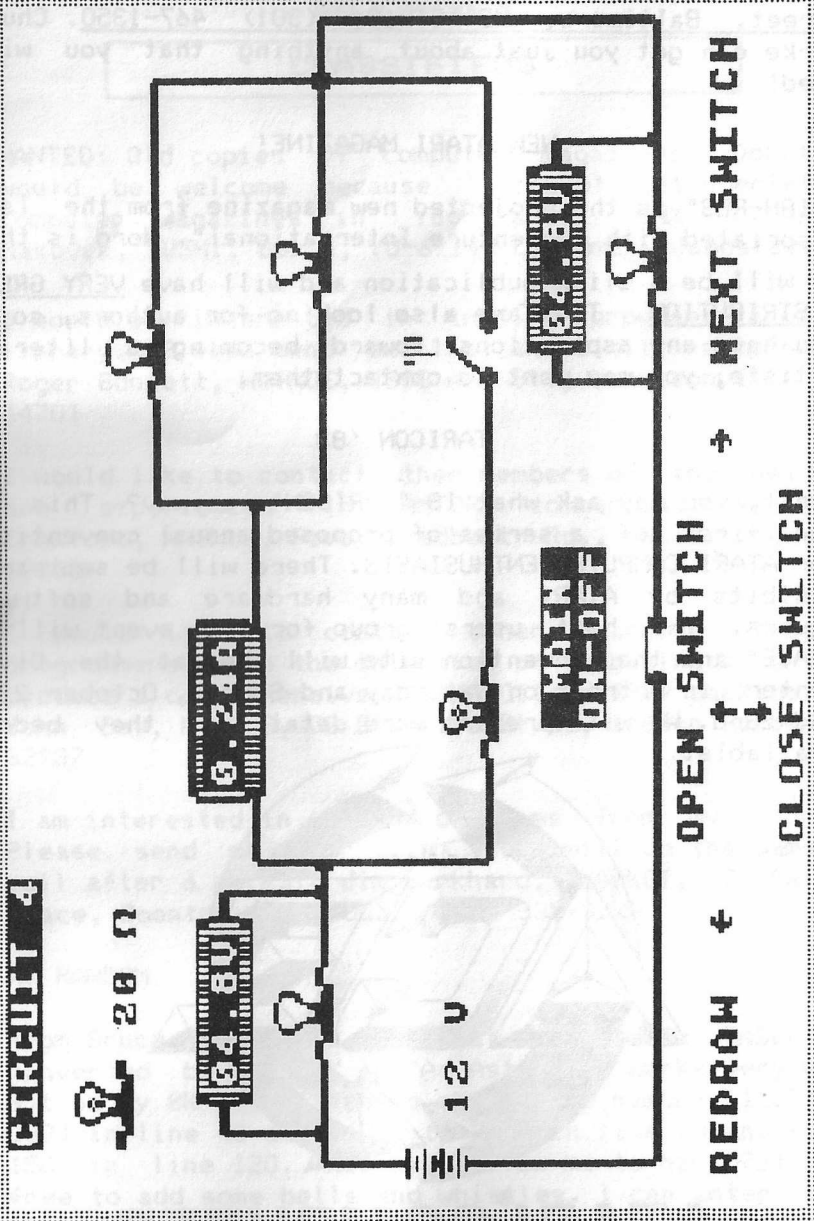
"HIGH-RES" is the projected new magazine from the lads associated with "Adventure International". Word is that it will be a slick publication and will have VERY GREAT DISTRIBUTION. They are also looking for authors, so if you have any aspirations toward becoming a literary artiste, you may want to contact them.

TARICON '83

First, you may ask what IS "TARICON" anyway? This is the first of a series of proposed annual conventions for ATARI COMPUTER ENTHUSIASTS. There will be seminars, exhibits by ATARI and many hardware and software houses. The host users' group for this event will be "MACE" and the convention site will be at the Civic Center in Detroit on Saturday and Sunday, October 22nd and 23rd. We will present more details as they become available.

ATARI® COMPUTER ENTHUSIASTS

Screen dump of 'CIRCUIT LAB' using Macrotronics' Screen Printer Interface & Driver program. Watch for a review in the next issue!



CW SYSTEM PROGRAM
by Martin Schick, KA4IWG

Operaion:

This program series of modules contain the main BASIC program and three machine language subprograms. Although the listing shows entries of these subprograms from cassette, you may have to modify them for entry from diskette. The ATARI editor-assembler will be necessary to enter the assembly code subroutines.

Once the final subprogram is loaded, the menu screen should appear. This screen will allow the user to select from the following modes: RECEIVE, TRANSMIT, SET SPEED, RANDOM CODE PRACTICE, LOGGING CALLS or OUTPUTTING THE LOG. When the command is entered, the screen will change to the mode chosen. To leave all other screen modes except LOGGING CALLS and CHANGING SPEED, it is necessary to strike a key to return to the menu. This will allow the user time to finish reading the output before it is cleared. In the TRANSMIT, RECEIVE and PRACTICE modes, the subprograms must first be stopped using the " A " key. This key will stop the code processing but will not clear the screen and return to the menu until another key is pressed.

If the RECEIVE mode was chosen, the screen will be cleared and then an asterisk will appear. This will show that the system is operational. The system will adjust itself to the speed of the code being sent. If it is a good signal and the code is being sent properly, the routine will work. The routine uses PORT 1 and looks at the first four pins.* An interface such as the "Ad Astra..." unit or the Kantronics "THE INTERFACE" will operate this routine. To leave the receive mode, enter the " A " key, then ANY other key.

If the TRANSMIT mode was chosen, the speed value is checked. If it is found to be zero or greater than 10, the system will ask the user for the code rate. The code rate has not been calibrated, so the values from 1 to 10 are used. The higher the number, the slower the code speed. The speed value is stored in a volatile portion of memory and therefore, may be lost from time to time. For this reason, the routine will occasionally ask for the speed. Once the speed has been entered, the screen is cleared and the transmit screen appears. The user can now send code with the keyboard. This routine has no buffer at this time, so only one character at a time is sent. This also means that any mistyped key will be sent. There are several SPECIAL FUNCTION KEYS listed at the end of this article.** To leave this mode, press the " A " key and any other key.

If the RANDOM CODE PRACTICE was chosen, the system will ask the user for a speed as described in the transmit section. Once this is entered, the screen is cleared and the random code practice screen appears. To start the code, strike ANY key. This will start the code to be sent in five character groups. To stope this routine, kit the " A " key. The text can then be checked before clearing the screen by entering another key.

The SPEED routine was described in the TRANSMIT section.

The LOG ROUTINE will clear the screen and then ask for the call to enter. When the call has been entered, the current log is checked for duplicate calls. If the call is a duplicate, there will be a warning, the call will not be stored, and the routine will return to the menu. If the routine senses that the computer is running low on memory, there will be a warning, though the system will contine to function for some time.

The LOG OUTPUT routine will dump all of the calls in memory to the screen. At this time there is no support for a printer. Once the calls have been dumped, the memory is re-initiated and the log zeroed. Thus, if memory becomes low, the log can be dumped and the system restarted. To leave this routine, strike ANY key.

The EXIT routine will bring the user back to BASIC. If the user wishes to restart the system, typing <RUN> will bring up the menu screen without reloading the subprograms. To reactivate the cursor while in BASIC, use the <BREAK> key. If the <SYSTEM RESET> key is depressed, the program will lose it's pointers. If <RUN> is then issued, the program will signal for loading the machine language subprograms. This is not necessary if they have already been loaded. When the signal to load the programs is given, try using the <BREAK> key and enter <RUN> again. This should reinitialize the pointers and the system will enter the menu mode.

* Pin 1 - CW to Computer
Pin 4 - CW out to TU
Pin 8 - Ground

** < - end of message
= - error
> - end of work
@ - wait

PROGRAM LISTINGS START ON PAGE 35

TWO "BIGGIES" FROM MACROTRONICS!
by Jack McKirgan II, WD8BNG

Back in March I received a call from Donna Burt, advertising manager of Macrotronics, Inc. Donna was very enthused about the new "TERMINALL" T4" in production by that company for the ATARI Computer System. She asked me if I would like to review one of the first production versions of the unit and before she could take another breath I leaped on the opportunity!

The unit arrived about 10 days before the Dayton Hamvention and because of the preparations being made at that time, I wasn't able to conduct full tests before that event, 'though we did take the unit with us to demonstrate it to prospective members of the net. I won't take up too much time describing the hardware..... an itemized listing with photos can be found on the following pages. I will tell you that the hardware does work as specified and that it was run side-by-side with a highly-touted (and very expensive) dedicated RTTY terminal and the Macrotronics "TERMINALL T4" kept right up with it! This is a good indication of a well-engineered piece of equipment. On the air tests of the demodulator indicate that it is VERY SENSITIVE and VERY SELECTIVE. Because of the selectivity, I found that I could print signals that were in the mud and surrounded by other stations... a feature that is quite heartening after trying to use a Kantronics "The Interface" on the crowded 40 meter band with little success.

As in almost any situation, Newton's Law of Tit for Tat applies here and it points out the only weakness in the hardware that I could find: High selectivity requires that both the sending and receiving stations be very stable! Even a very slight drift can send your hand to the VFO dial to "touch-up" the receiver a little. While this is nothing unusual, the only tuning indicator on the TU is a meter that must be tuned to peak on "mark" and it will deflect very little when properly tuned. I never did "master" the method on shifts other than 170 Hz. I did attach the user port to my YO-100 'scope and tuning was easily acheived using the ellipse-target

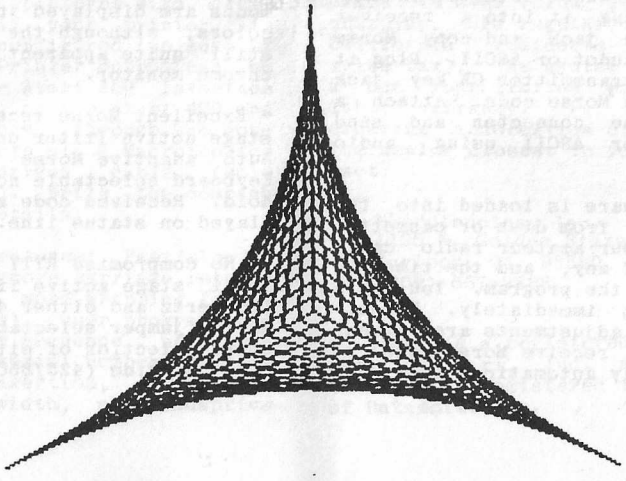
method. However, not everyone has a 'scope available in the shack and a tuning-eye or bar graph readout would be a nice addition to the unit. Changing between the "alternate" shift of either 425 or 850 Hz. is accomplished under software control, although you must select which if the "alternate" shifts you want by changing jumpers on the PC board of the TU.

The software is supplied on both disk and cassette with each TU and a 32K system is minimum. A 48K machine will leave about 28K of dynamic memory (5K more if you use tape as DOS is not resident). By "dynamic" memory, I mean that it is reconfigurable and there are no restrictions in the number of characters allotted to each of the 16 message buffers as long as the total does not exceed the free memory. Except for one bug, the software seems to be magnificent! It even works with the XL operating system! ALL of the most wanted features are included in the software including disk I/O and program transfer capabilities with 6, 7 or 8 bit ASCII codes accepted. (Yes you CAN transfer all of those special ATASCII control characters and graphics characters!) The program works in CW/RTTY/ASCII modes and the CW receive algorhythm seems to track some pretty sloppy fists! The options available are bewildering for there are no less than three pages of commands that can be called by using combinations of <START>, <SELECT> and <OPTION> keys with standard keys. As an example, <START>-I will send a CW ID immediately during transmit whereas <SELECT>-I will tell the program to send the ID automatically every six minutes! Consider the number of key-combinations... you can see that at first it can be bewildering! Macrotronics has certainly gone all-out to introduce the user to the capabilities of the combo! There is even a section in the 109 page manual to tell you what to do if you want to get on-line immediately and don't want to read the whole manual! It is impossible in the space of this article to describe all of the features of the unit and it's software... check the features in the itemized listing and try to imagine at least two options for each function!

There was one bug that I hadn't noticed until one of our members said that he thought he had found it! I ran

a series of on-the-air tests with Bob, KA8HCG, and we did confirm that the ASCII/RTTY conversion table was messed up a bit and that several of the punctuation marks were transposed, i.e. hitting a period would send a slant bar etc. (No one had mentioned this previously on the air during my evaluations... probably because RTTY operators expect some strange-looking print due to operator habits!) At any rate, this was ONLY experienced during RTTY operations and not ASCII or CW. I did call Nate Olson, a member of the net and representative of Macrotronics, and Nate took the information that I gave and confirmed it with a system at the company. Nate caled me back and assured me that the bug would be fixed on future releases of the software and want's present owners to BE AWARE THAT MACROTRONICS WILL REPLACE THE INITIAL SOFTWARE RELEASE WITH THE CORRECTED VERSION. This release should be available now and if you contact Nate or Donna you will be able to obtain the new version.

Having been one the first owners of a Macrotronics unit... the M-80 for the TRS-80 (about 6 years ago!), I KNEW that this would be a quality product and the software would be fantastic! Macrotronics didn't let me down! I have heard a lot of promises made by other hardware and software houses in the last few months, but this combination is here, available and works! It is not inexpensive... $499 + $4 shipping, but it is well worth it if you are going to be serious about RTTY/ASCII/CW with the ATARI. I personally am going to buy one!



NEW PRODUCT ANNOUNCEMENT

Macrotronics, Inc has announced the introduction of TERMINALL T4, an integrated hardware and software system which converts an Atari 400*, Atari 800* or Atari 1200* computer into a state of the art radio communications terminal. This product is essentially a radio modem and allows amateur radio operators to send and receive Morse, Baudot and ASCII codes over a radio. It also allows displaying and printing a variety of news, weather and other wire services which are broadcast over short-wave radio.

TERMINALL includes all the necessary computer interfacing, audio demodulating, AFSK tone generating and transmitter keying hardware integrated in one cabinet. This reduces equipment interconnection to a minimum and allows the operator to be on the air receiving and transmitting Morse or Baudot or ASCII in minutes. Plug it into a receiver headphone jack and copy Morse code, Baudot or ASCII. Plug it into a transmitter CW key jack and send Morse code. Attach a microphone connector and send Baudot or ASCII using audio tones.

The software is loaded into the computer from disk or cassette. Enter your amateur radio callsign, if any, and the time to initiate the program. You begin receiving immediately. No settings or adjustments are necessary to receive Morse code -- it's fully automatic. Text may be typed in a split screen format while receiving or transmitting.

Some of the features of TERMINALL T4 are:

* Multi-level Displays: Edit Window on top to enter transmit text or program messages. Status Window shows operating parameters, prompts and error messages. History Window displays received and transmitted text in chronological order. Review Window allows examining and editing historical text while receiving or transmitting.

* Cursor editing: Use the cursor control keys to compose, insert, delete or write over any text to be transmitted, any preprogrammed messages or any received text. You can edit received text, such as W1AW bulletins, before or after saving to a file.

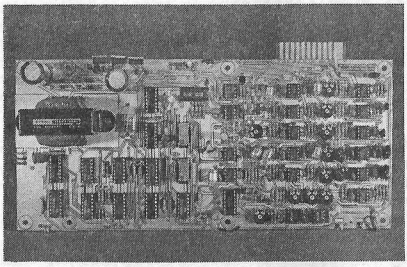
* Messages and received text may be saved to disk or cassette. Disk files are compatible with most word processors including Text Wizard*. BASIC programs may be transferred over the radio.

* Built-in backup routine saves all user selected options (such as callsign, modes, messages, etc.) as defaults.

* Receive, transmit and break modes are displayed in different colors, although the modes are still quite apparent on a monochrome monitor.

* Excellent Morse reception: Six stage active filter demodulator. Auto adaptive Morse algorithm. Keyboard selectable noise threshold. Received code speed displayed on status line.

* No compromise RTTY reception: Multi stage active filters for 170 Hertz and either 425 or 850 Hertz (jumper selectable). Keyboard selection of either Narrow (170) or Wide (425/850) shift.



HARDWARE

* Hardware clock which maintains accurate time during all operations, including Disk I/O. User programmable time/date format.

* ASCII capabilities: Select even/odd/no parity. Select 6, 7 or 8 data bits. Select 75 or 110 baud. You may send and receive the full ASCII character set, including control codes.

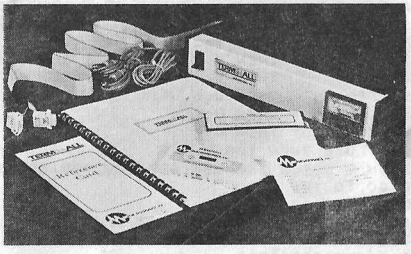
* Multiple user - defined WRU: For each of four WRU functions, the operator can select any combination of (1) Initiate sequence, (2) Terminate sequence (including none or timeout), (3) What to transmit back (if anything -- including ID in any mode, any message, any serial number and time/date), and (4) Whether to save on disk or cassette or not at all. WRU functions work in all modes (Morse, Baudot or ASCII).

* Buffered ASCII parallel printer output: Select edited historical text, all text or WRU activated ("AUTO START") text. You may print pictures with overprinting if your printer is properly configured (no auto LF on CR). Printer output can be through the Atari 850 Interface module or, on the Atari 400 and 800 computers, through the controller ports via a Macrotronics printer driver cable (sold separately). Write the company for more details.

* Other features: Fast/slow/no diddle, ignore carriage returns on receive, word wrapping (won't split words), user programmable end of line sequence, user programmable serial number and time format insertion, adjustable carriage width, auto adaptive transmit delay, Break mode. Selectable from the keyboard: Baud rate, shift, CW ID keying, unshift-on-space, signal invert, Morse/RTTY toggle, Morse transmit speed.

* Flexible interfacing: Built in: Separate CW and RTTY active filter demodulators, crystal controlled AFSK, separate relays for keying CW and PTT, solid state FSK driver, scope outputs, 60 mil loop opto-isolated interconnect, Serial (RS232 compatible) IN and OUT, hand-key input, side-tone output, jumper selectable 110/220 volt AC power supply and jumper selectable 425 or 850 Hertz wide shift.

* TERMINALL T4 requires an Atari 400 or Atari 800 computer with a minimum of 32 K RAM, or an Atari 1200 computer, with one disk drive or a cassette recorder.



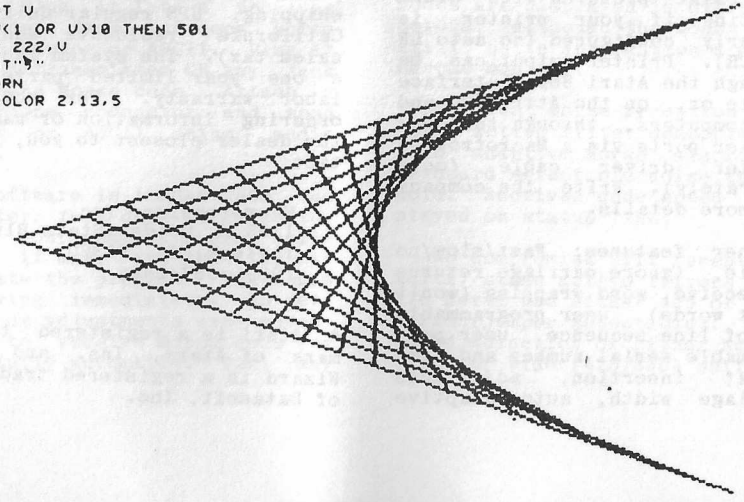
PACKAGE CONTENTS

Package includes software on cassette and diskette, assembled and tested hardware and extensive instruction manual. List price is $499 (plus $4.00 for shipping, UPS regular delivery, California residents add 6% sales tax). The system includes a one year limited parts and labor warranty. For complete ordering information or name of the dealer closest to you, contact:

Macrotronics, Inc.
1125 N. Golden State Blvd
Turlock, CA. 95380
(209) 667-2888

* Atari is a registered trademark of Atari, Inc. and Text Wizard is a registered trademark of Datasoft, Inc.

```
1 POKE 752,1:TOP=PEEK(741)+PEEK(742)*256
2 STC=TOP:Y=PEEK(TOP):IF Y=104 THEN 90
3 TOP=TOP-512
4 STC=TOP
5 HI=INT(TOP/256)
7 LO=TOP-HI*256
9 POKE 741,LO:POKE 742,HI
10 BIF=1536
11 GOSUB 3100
13 T=32:POKE 1772,T:POKE 1776,T:POKE 1780,T:T=96:POKE 1775,T:POKE 1779,T:POKE 1783,T
17 BIF=TOP+4:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1784,LO:POKE 1785,HI
21 BIF=TOP+8:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1786,LO:POKE 1787,HI
25 BIF=TOP+95:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1777,LO:POKE 1778,HI
29 BIF=TOP+150:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1781,LO:POKE 1782,HI
33 BIF=TOP+205:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1788,LO:POKE 1789,HI
35 BIF=TOP+256+56:HI=INT(BIF/256):LO=BIF-256*HI:POKE 1770,LO:POKE 1771,HI
37 BIF=TOP+256+136:HI=INT(BIF/256):LO=BIF-HI*256:POKE 1773,LO:POKE 1774,,HI
39 BIF=TOP+256+19:HI=INT(BIF/256):LO=BIF-HI*256:POKE 1790,LO:POKE 1791,HI
40 BIF=TOP
42 GOSUB 3100
44 BIF=TOP+256
46 GOSUB 3100
90 DIM V$(1),C$(6),B(7),E$(6)
100 SETCOLOR 2,9,4:PRINT"}"
110 POSITION 10,11:PRINT"CW SYSTEM MENU"
120 POSITION 10,13:PRINT" ENTER LETTER FOR ROUTINE"
130 POSITION 10,16:PRINT"       T - TRANSMIT"
140 POSITION 10,17:PRINT"       R - RECIEVE"
150 POSITION 10,18:PRINT"       S - SPEED"
160 POSITION 10,19:PRINT"       P - RANDOM CODE"
170 POSITION 10,20:PRINT"       E - EXIT"
172 POSITION 10,21:PRINT"       L- LOG IN CALL"
174 POSITION 10,22:PRINT"       O - OUTPUT LOG"
190 V=0
200 PRINT "COMMAND ";:INPUT V$
201 IF V$="T" THEN V=1
202 IF V$="R" THEN V=2
203 IF V$="S" THEN V=3
204 IF V$="P" THEN V=4
205 IF V$="E" THEN V=5
206 IF V$="L" THEN V=6
207 IF V$="O" THEN V=7
209 IF V=0 THEN 100
210 ON V GOSUB 300,400,500,600,700,800,900
220 GO TO 100
300 V=PEEK(222)
301 PRINT "{":SETCOLOR 2,12,5
310 IF V<1 OR V>10 THEN GOSUB 501
350 V=USR(TOP+256)
355 GOSUB 950
360 RETURN
400 PRINT "}":SETCOLOR 2,9,4
401 V=USR(TOP)
402 GOSUB 950
410 RETURN
500 SETCOLOR 2,2,5
501 PRINT"}":POSITION 2,20:PRINT"ENTER A VALUE FROM 1 TO 10 FOR SPEED"
510 INPUT V
515 IF V<1 OR V>10 THEN 501
520 POKE 222,V
521 PRINT"}"
530 RETURN
600 SETCOLOR 2,13,5
```



```
601 GOSUB 501
620 BIF=TOP+256+156:HI=INT(BIF/256):LO=BIF-256*HI
630 L=PEEK(1790):H=PEEK(1791):POKE 1790,LO:POKE 1791,HI
631 POKE 222,U:POKE 214,0
640 U=USR(TOP+256)
650 POKE 1790,L:POKE 1791,H
651 GOSUB 950
660 RETURN
700 PRINT "{CLEAR}"
701 SETCOLOR 2,9,4
702 STOP
800 SETCOLOR 2,1,6:PRINT"{CLEAR}":POSITION 10,10:PRINT"ENTER CALL SIGN" ;:INPUT C$
801 Y=FRE(0):IF Y<2048 THEN PRINT"{CLEAR}TIME TO OUTPUT CALLS"
802 FOR Y=1 TO 300:NEXT Y
803 Y=LEN(C$):IF Y=6 THEN 807
805 FOR U=Y+1 TO 6:C$(U,U)="\":NEXT U
807 IF ASC(C$(6,6))<64 THEN 811
809 E$=C$:C$(1,1)=C$(6,6):C$(2)=E$(1,5): GO TO 807
811 NUM=VAL(C$(6,6))
813 C1=ASC(C$(1,1)):C2=ASC(C$(2,2)):C3=ASC(C$(3,3)):C4=ASC(C$(4,4)):C5=ASC(C$(5,5))
815 C1=C1-64:C2=C2-64:C3=C3-64:C4=C4-64:C5=C5-64
817 C2=C2*32:C3=C3*1024:C4=C4*32768:C5=C5*1048576:C6=NUM*33554432
819 C=C1+C2+C3+C4+C5+C6
821 FOR U=1 TO 7:B(U)=INT(C-INT(C/16)*16):C=C/16:NEXT U
823 Y=INT((TOP-STC)/7)-1
825 FOR U=0 TO Y
827 FOR T=1 TO 7
829 IF PEEK(STC+(7*U(T-1)))<>B(T) THEN 837
831 NEXT T
832 SETCOLOR 2,4,5
833 PRINT "{CLEAR}":POSITION 10,10:PRINT"DUPE CALL":FOR T=1 TO 300:NEXT T:RETURN
837 NEXT U
840 STC=STC-7:HI=INT(STC/256):LO=STC-HI*256
842 POKE 741,LO:POKE 742,HI:FOR U=0 TO 6:POKE STC+U,B(U+1):NEXT U
844 RETURN
900 HI=INT(TOP/256):LO=TOP-HI*256:POKE 741,LO:POKE 742,HI
901 SETCOLOR 2,6,4:PRINT"{CLEAR}"
902 IF STC>=TOP THEN 950
903 FOR U=0 TO 6:B(U+1)=PEEK(STC+U):NEXT U
905 B=B(1)+B(2)*16+B(3)*256+B(4)*4096+B(5)*65536+B(6)*1048576+B(7)*16777216
907 FOR U=1 TO 5:B(U)=(B-INT(B/32)*32)+64:B=INT(B/32):NEXT U
909 B(6)=INT(B)+48
910 Y=3
911 FOR U=1 TO 6:IF B(7-U)=28 THEN 917 :NEXT U
913 GO TO 919
917 Y=U-1
919 FOR U=1 TO Y
921 D=B(6):FOR G=6 TO 2 STEP -1:B(G)=B(G-1):NEXT G:B(1)=D
923 NEXT U
927 FOR U=1 TO 6
928 IF B(U)=92 THEN B(U)=32
929 NEXT U
930 FOR U=1 TO 6:PRINT CHR$(B(U));:NEXT U
931 PRINT
933 STC=STC+7:GOTO 902
950 PRINT:PRINT" STRIKE ANY KEY WHEN READY"
951 Y=PEEK(764):IF Y=255 THEN 951
955 POKE 764,255
960 RETURN
3100 TRAP 3260
3110 OPEN #3,4,0,"C:"
3120 GET #3,X
3130 GET #3,X
3140 GET #3,X
3150 GET #3,Y
3160 ADSTART=256*Y+X
3170 GET #3,X

3180 GET #3,Y
3190 ADEND=256*Y+X
3200 ADCUR=ADSTART
3210 GET #3,X
3220 POKE BIF,X
3230 ADCUR=ADCUR+1
3231 BIF=BIF+1
3240 IF ADCUR<=ADEND THEN GOTO 3210
3250 GO TO 3140
3260 CLOSE #3
3270 RETURN
```

```
            SIGIN=$D300
            POTCTL=$D302
            POTDAT=SIGIN
            ATRACT=$4D
            NOISE=$01
            LETT=$2FC

            TEMP=$D6
            DIT=TEMP+1
            DOT=DIT+1
            DASH=DOT+1
            GAP=DASH+1
            FLAG=GAP+1
            FAH=FLAG+1

            CWJSR=$6F0
            CONJSR=CWJSR+4
            STRJMP=CONJSR+4
            PAWJMP=STRJMP+2
            OUTJMP=PAWJMP+2

            STRTAD=$4000
            CNTDWN=$630
            TAB=$600

            *=STRTAD
4000  68                  PLA                     ;STRIP POINTER FROM CALL
4001  20F006              JSR     CWJSR           ;INDIRECT JUMP TO INITIALIZE
      ;
      ;
      ;THIS SECTION IS THE MAINLINE... IT WAITS UNTIL A CARRIER IS
      ;DETECTED AND THE TIMES THE LENGTH OF THE SIGNAL... IF THE
      ;CARRIER IS OFF TOO LONG IT ASSUMES A CHARACTER HAS BEEN SENT
      ;AND WILL TRY TO DO A CONVERSION...THIS TIMING IS DONE BY AN
      ;INTERRUPT ROUTINE LOCATED ON PAGE SIX
4004  A0FF        START   LDY     #$FF
4006  A6DA                LDX     GAP             ;GET LAST LETTER SPACE
4008  84D6        PAUSE   STY     TEMP            ;START SOFTWARE TIMER
400A  CC00D3      MARK    CPY     SIGIN           ;CHECK FOR CARRIER
400D  D00F                BNE     SPACE           ;BRIF CARRIER ON
400F  E4D6                CPX     TEMP            ;CHECK FOR LETTER SPACE
4011  90F7                BCC     MARK            ;BRIF NO LETTER SPACE
4013  20F406              JSR     CONJSR          ;INDIRECT JUMP TO OUTPUT ROU
4016  C0FF                CPY     #$FF            ;CHECK FOR END
4018  F001                BEQ     GOON            ;BRIF NOT END
401A  60                  RTS                     ;DONE
401B  6CF806      GOON    JMP     $(STRJMP)       ;INDIRECT JUMP TO START
401E  84D6        SPACE   STY     TEMP            ;RESTART SOFTWARE TIMER
4020  CC00D3      SIGON   CPY     SIGIN           ;CHECK FOR CARRIER
4023  D0FB                BNE     SIGON           ;BRIF CARRIER STILL ON
      ;
      ;
      ;THIS SECTION TAKES THE VALUE OF THE TIMER IN THE INTERRUPT
      ;ROUTINE AND USES IT TO MEASURE THE CARRIER LENGTH... IF
      ;THE VALUE IS SHORTER THAN THE NOISE LENGTH, THE CARRIER IS
      ;IGNORED... IF THE LENGTH IS MORE THAN TWICE THE VALUE IN THE
      ;DIT REGISTER, IT IS CHECKED TO BE A DAH... IF THE VALUE IS
      ;NOT AS LONG AS THE DAH REGISTER LENGTH IT IS ASSUMED TO BE A
      ;DIT... THE LENGTH IS THEN AVERAGED WITH THE APPROPRIATE REG
      ;ISTER TO KEEP A RUNNING AVERAGE... THE CHARACTER REGISTERS
      ;ARE THEN SHIFTED AND BIT ONE OF THE APPROPRIATE REGISTER IS
      ;SET
4025  A5D6                LDA     TEMP            ;GET TIMER VALUE
4027  49FF                EOR     #$FF            ;COMPLEMENT THE VALUE
4029  C901                CMP     #NOISE          ;CHECK FOR NOISE SPIKE
402B  90DD                BCC     MARK            ;BRIF NOISE SPIKE
402D  4A                  LSR       A             ;DIVIDE BY TWO
402E  C5D7                CMP     DIT             ;CHECK FOR LENGTH
4030  B016                BCS     HASH            ;BRIF LENGTH IS LONGER THAN
4032  0A                  ASL       A             ;CHANGE BACK
4033  18          BIT     CLC                     ;GOT A DIT SO
4034  65D7                ADC     DIT             ;ADD TO DIT LENGTH
4036  6A                  ROR       A             ;AND DIVIDE BY 2 TO AVERAGE
4037  85D7                STA     DIT             ;STORE RUNNING AVERAGE
4039  0A                  ASL       A             ;DOUBLE THE LENGTH
403A  85DC                STA     FAH             ;STORE FOR DAH LENGTH
403C  A5D8                LDA     DOT             ;GET DIT REGISTER
403E  0A                  ASL       A             ;ROTATE ONE SPOT
403F  0901                ORA     #$01            ;AND MASK BIT
4041  85D8                STA     DOT
4043  06D9                ASL     DASH            ;ROTATE DAH REGISTER
4045  6CFA06              JMP     $(PAWJMP)       ;INDIRECT JUMP TO PAUSE
4048  0A          HASH    ASL       A             ;RESTORE ORIGINAL DATA
4049  C5DC                CMP     FAH             ;CHECK FOR DAH LENGTH
```

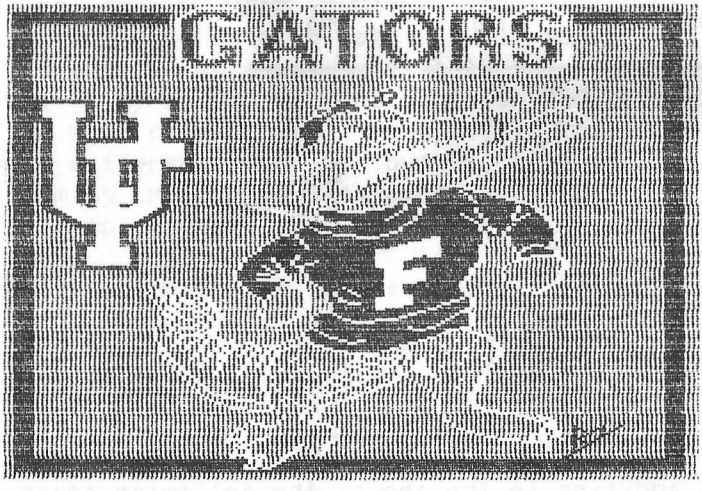
```
404B    90E6          BCC    BIT                ;BRIF IF LENGTH INDICATES DIT
404D    18            CLC                       ;ADD LENGTH TO DAH LENGTH
404E    65DC          ADC    FAH                ;AND DIVIDE BY TWO
4050    6A            ROR      A                ;TO STORE RUNNING
4051    85DC          STA    FAH                ;AVERAGE
4053    A5D9          LDA    DASH               ;GET DAH REGISTER
4055    0A            ASL      A                ;SHIFT ONE BIT
4056    0901          ORA    #$01               ;AND ADD MASK BIT
4058    85D9          STA    DASH
405A    06D8          ASL    DOT                ;SHIFT DIT REGISTER
405C    6CFA06        JMP    $(PAWJMP)          ;INDIRECT JUMP TO PAUSE
    ;
    ;
    ;THIS SECTION IS THE INITIALIZATION ROUTINE...IT FINDS THE LOCATION
    ;OF THE DISPLAY LIST AND CHANGES EVERY OTHER LINE FOR INTERRUPT...
    ;THIS ASSUMES GRAPHICS 2 IS USED... IT THEN SETS THE INTERRUPT
    ;VECTOR FOR THE TIMER LOCATED ON PAGE 6... THIS TIMER IS USED IN
    ;STEAD OF THE SOFTWARE CLOCKS TO GET SPEEDS FASTER THAN 1/60TH OF
    ;A SECOND... THE ROUTINE THEN SETS THE PIA FOR READING OF ALL PINS
    ;ON PORTS 1 AND 2... FINALLY IT STORES TYPICAL VALUES IN THE LENGTH
    ;REGISTERS TO START WITH
405F    AC3002    CW    LDY    $0230              ;POINT TO DISPLAY
4062    AE3102          LDX    $0231              ;ADDRESS
4065    84D8            STY    DOT                ;AND STORE IT IN
2067    86D9            STX    DOT+1              ;A HANDY LOCATION
4069    A006            LDY    #$06               ;SET COUNTER
406B    A982            LDA    #$82               ;LOAD INTERUPT WORD FOR GRAPHIC
406D    91D8      INT   STA    (DOT),Y            ;STORE IN DISPLAY LIST
406F    C8              INY                       ;JUMP TWO LINES
4070    C8              INY                       ;IN DISPLAY LIST
4071    C01C            CPY    #$1C               ;CHECK FOR END OF LIST
4073    90F8            BCC     INT               ;BRIF NOT END
4075    A230            LDX    #CNTDWN&$00FF      ;LOAD ADDRESS
4077    A006            LDY    #CNTDWN/256        ;OF INTERUPT ROUTINE
4079    8E0002          STX     $0200             ;AND STORE IN
407C    8C0102          STY     $0201             ;PROPER LOCATIONS
407F    A980            LDA    #$80               ;LOAD INTERUPT MASK
4081    0D0ED4          ORA     $D40E             ;MASK CONTROL WORD
4084    8D0ED4          STA     $D40E
4087    20DAE6          JSR     $E6DA             ;OS ADDRESS FOR PIA SETUP
408A    A950            LDA    #$50               ;STORE INITIAL DIT VALUE
408C    85D7            STA     DIT
408E    0A              ASL      A                ;DOUBLE LENGTH
408F    85DC            STA     FAH               ;STORE INITIAL DAH VALUE
4091    49FF            EOR    #$FF               ;COMPLEMENT

4093    85DA            STA     GAP               ;AND STORE INITIAL LETTER SPAC
4095    60              RTS
    ;
    ;
    ;THIS SECTION DOES THE CONVERSION FROM CODE TO ASCII... WHEN
    ;ENTERED, IT HASHES THE DIT AND DAH REGISTERS IN A WAY TO GIVE
    ;UNIQUE CODES FOR EACH CHARACTER... IT THEN CHECKS IF THIS VALUE
    ;IS ZERO... IF IT IS, THE ROUTINE ASSUMES THE GAP LENGTH HAS BEEN
    ;ACCEEDED AND A SPACE IS SENT ONLY IF THE LAST CHARACTER SENT WAS
    ;NOT A SPACE... IF THE HASHING IS NONE ZERO, THE ROUTINE DOES A
    ;TABLE LOOK-UP TO FIND THE MATCHING CODE... THESE ARE IN ORDER OF
    ;THEIR ASCII POSITIONS... THE TABLE OFFSET IS ADDED TO THE ASCII
    ;OFFSET AND THE VALUE OUTPUT... IF THE CODE IS NOT LOCATED, THE
    ;ERROR CHARACTER IS OUTPUT INSTEAD
4096    A5DC      CONVT LDA    FAH                ;GET DAH LENGTH
4098    49FF            EOR    #$FF               ;COMPLEMENT IT
409A    85DA            STA     GAP               ;STORE NEW SPACE
409C    CCFC02          CPY     LETT              ;CHECK KEYBOARD
409F    D039            BNE     KEY               ;BRIF KEY STRUCK
40A0    A5D8            LDA     DOT               ;GET DIT VALUES
40A3    0A              ASL      A                ;SHIFT
40A4    18              CLC                       ;AND ADD THE
40A5    65D9            ADC     DASH              ;DAH VALUES
40A7    D00C            BNE     ZERO              ;BRIF VALUE IS NON-ZERO
40A9    C4DB            CPY     FLAG              ;CHECK FLAG FOR ONE SPACE ALRE
40AB    D001            BNE     MOVE              ;BRIF LAST CHAR WAS NOT SPACE
40AD    60              RTS                       ;GO BACK
40AE    84DB      MOVE  STY     FLAG              ;STORE A VALUE IN FLAG
40B0    A920            LDA    #$20               ;LOAD SPACE CHAR
40B2    6CFC06          JMP    $(OUTJMP)          ;INDIRECT JUMP TO OUTPUT
40B5    A22E      ZERO  LDX    #$2E               ;LOAD TABLE COUNT
40B7    DD0006    NEXT  CMP     TAB,X             ;LOOK FOR CHAR CODE
40BA    F005            BEQ     $ASCII            ;BRIF FOUND
40BC    CA              DEX
40BD    10F8            BPL     NEXT              ;BRIF STILL LOOKING
40BF    A233      ERROR LDX    #$33               ;LOAD ERROR CHARACTER
40C1    A900      ASCII LDA    #$00               ;REINITIALIZE THE
40C3    85D9            STA     DASH              ;DAH REGISTER
40C5    85D8            STA     DOT               ;AND DIT REGISTER
```

```
40C7    85DB            STA     FLAG            ;RESET SPACE FLAG
40C9    8A              TXA                     ;GET ASCII CHARACTER OFFSET
40CA    18              CLC                     ;AND ADD TO
40CB    692B            ADC     #$2B            ;BASE VALUE
        ;
        ;
        ;THIS ROUTINE SENDS THE CHARACTER TO THE SCREEN... THEN IT CHECKS
        ;IF A LOGO KEY WAS ENTERED... IF IT WAS, THE PROGRAM ENDS... IT
THEN SETS THE ATTRACT TIMER TO KEEP THE SCREEN FROM CHANGING
40CD    20A4F6  OUTPUT  JSR     $F6A4           ;JUMP TO OS OUTPUT ROUTINE
40D0    A0FF            LDY     #$FF            ;RESTORE Y
40D2    854D            STA     ATRACT          ;SET ATTRACT REGISTER
40D4    CCFC02          CPY     LETT            ;CHECK KEYBOARD REGISTER
40D7    D001            BNE     KEY             ;BRIF KEY HIT
40D9    60              RTS
40DA    ADFC02  KEY     LDA     LETT            ;GET KEY CODE
40DD    8CFC02          STY     LETT            ;RESTORE KEY REGISTER
40E0    C927            CMP     #$27            ;CHECK IF KEY WAS A LOGO KE
40E2    F001            BEQ     ESC             ;BRIF IT WAS
40E4    60              RTS
40E5    88      ESC     DEY
40E6    A97F            LDA     #$7F            ;LOAD MASK
40E8    2D0ED4          AND     $D40E           ;MASK INTERUPT
40EB    8D0ED4          STA     $D40E           ;STORE IT
40EE    60              RTS                     ;SET END AND RETURN
        ;
        ;

        ;THIS IS THE INTERRUPT ROUTINE... IT TAKES CHECKS THE LOCATION
        ;TEMP FOR ZERO... IF NONE ZERO, IT IS DECREMENTED... THIS IS
        ;A COUNT-DOWN CLOCK
        *=CNTDWN
0630    48              PHA                     ;STORE ACCUM ON STACK
0631    A900            LDA     #$00            ;ZERO ACCUM
0633    C5D6            CMP     TEMP            ;CHECK FOR TIMEOUT
0635    F002            BEQ     OUT             ;BRIF TIMER DONE
0637    C6D6            DEC     TEMP            ;DECREMENT TIMER
0639    68      OUT     PLA                     ;RESTORE ACCUM
063A    40              RTI
        ;
        ;
        ;THIS IS THE TABLE OF CODE LETTERS USED IN LOOKING FOR THE ASCII
        ;CHARACTER RECIEVED
        *=TAB
0600    FF4B                    .BYTE $FF,$4B,$2D,$69,$2C,$1F,$2F,$37
0602    2D69
0604    2C1F
0606    2F37
0608    3B3D                    .BYTE $3B,$3D,$3E,$2E,$26,$22,$20,$46
060A    3E2E
060C    2622
060E    2046
0610    5434                    .BYTE $54,$34,$28,$79,$72,$36,$05,$16
0612    2879
0614    7236
0616    0516
0618    140A                    .BYTE $14,$0A,$02,$1C,$08,$1E,$06,$17
061A    021C
061C    081E
061E    0617
0620    091A                    .BYTE $09,$1A,$03,$04,$07,$18,$11,$0C
0622    0304
0624    0718
0626    110C
0628    0E01                    .BYTE $0E,$01,$0D,$1D,$0B,$15,$13,$12
062A    0D1D
062C    0B15
062E    1312
```

```
                POTDAT=$D300
                SIGIN=POTDAT
                POTCTL=POTDAT+2
                CLOCK=$220
                RANDOM=$D20A
                TABST=$640
                CODEST=$66F

                TEMP=$D6
                CHAR=$DD
                SPEED=$DE

                JSRDLY=$6EC
                JMPSTR=$6FE

                      *=$6000
    6000    68                  PLA                 ;STRIP OFF COUNTER
;
;
;THIS SECTION SETS UP THE PIA FOR DATA OUT ON PORTS ONE AND TWO...
;AND THEN HOLDS IT IN THE OFF STATE
    6001    A938                LDA     #$38        ;COMMAND FOR DATA DIRECTION
    6003    A0FF                LDY     #$FF        ;COMMAND FOR DATA OUT
    6005    A23C                LDX     #$3C        ;COMMAND FOR DATA ADDRESS
    6007    8D02D3              STA      POTCTL     ;SET PIA FOR DATA DIRECTION
    600A    8C00D3              STY      POTDAT     ;SET DATA DIRECTION
    600D    8E02D3              STX      POTCTL     ;SET DATA ADDRESS
    6010    8C00D3              STY      SIGIN      ;BUT SHUT OFF FOR NOW
;
;
;NOW WE WAIT UNTIL A KEY IS STRUCK... IF IT IS A LOGO KEY WE STOP
;EVERYTHING... IF ITS A SPACE WE JUMP TO THE SPACE ROUTINE
    6013    A0FF        START   LDY     #$FF        ;LOAD MASK
    6015    CCFC02              CPY      LETT       ;CHECK FOR CHARACTER
    6018    F0F9                BEQ      START      ;BRIF NO CHARACTER
    601A    ADFC02              LDA      LETT       ;LOAD CHARACTER
    601D    8CFC02              STY      LETT       ;RESET REGISTER
    6020    20D8FC              JSR     $FCD8       ;OS ROUTINE FOR CLICK
    6023    C927                CMP     #$27        ;CHECK FOR LOGO KEY
    6025    F070                BEQ      RETURN     ;BRIF LOGO HIT
    6027    C921                CMP     #$21        ;CHECK FOR SPACE
    6029    F04B                BEQ      SPACE      ;BRIF SPACE BAR
    602B    A22E                LDX     #$2E        ;LOAD THE NUMBER OF LETTERS
;
;
;THIS IS THE LOOKUP ROUTINE... IT WILL TRY TO MATCH THE INTERNAL KEY
;CODE TO THE CODES IN THE TABLE... WHEN IT FINDS IT, THE OFFSET WILL
;BE THE ASCII OFFSET AND THE CODE OFFSET... IF IT CAN'T FIND IT, IT
;SIMPLY RETURNS
    602D    DD4006      LOOKUP  CMP      TAB,X      ;LOOK FOR MATCH
    6030    F006                BEQ      MORSE      ;BRIF MATCH
    6032    CA                  DEX
    6033    10F8                BPL      LOOKUP     ;KEEP LOOKING
    6035    6CFE06              JMP     $(JMPSTR)   ;NO MATCH-FORGET IT
;
;
;THIS ROUTINE ADDS THE ASCII BASE VALUE AND OUTPUTS THE CHARACTER
;... IT THEN LOADS THE CODE BYTE AND SHIFTS IT UNTIL IT LOCATES
;A CARRY WHICH ACTS AS A START BIT... THE NUMBER OF BITS LEFT IN THE
;WORD IS THE NUMBER OF CODE CHARACTERS IN THE LETTER
    6038    86DD        MORSE   STX      CHAR       ;SAVE CHACTER OFFSET
    603A    8A                  TXA
    603B    18                  CLC

    603C    692C                ADC     #$2C        ;ADD ASCII OFFSET
    603E    20A4F6              JSR      $F6A4      ;AND OUTPUT
    6041    A6DD                LDX      CHAR       ;RETRIEVE OFFSET
    6043    BD6F06              LDA      CODE,X     ;LOAD CODE CHARACTER
    6046    A207                LDX     #$07        ;LOAD BITS TO CYCLE
    6048    0A          STARTB  ASL        A        ;SHIFT THE CODE
    6049    CA                  DEX
    604A    90FC                BCC      STARTB     ;CHECK FOR START BIT
    604C    85DD                STA      CHAR       ;NOW SAVE TRUE CODE
;
;
;NOW WE SHIFT THE CODE INTO THE CARRY... IF THERE IS A CARRY, THEN
;WE OUTPUT A DAH BY SETTING THE COUNTS TO 3... IF NO CARRY, WE OUTPUT
;A DAH BY SETTING COUNTS TO 1... WE CONTINUE THIS UNTIL THE COMPLETE
;BYTE HAS BEEN SHIFTED
    604E    A5DD        NEXT    LDA      CHAR
    6050    0A                  ASL        A        ;SHIFT IT
    6051    85DD                STA      CHAR       ;SAVE IT
    6053    A001                LDY     #$01        ;LOAD DIT COUNT
    6055    9002                BCC      SEND       ;CHECK IF CODE WAS DAH
    6057    A003                LDY     #$03        ;LOAD DAH COUNT
;
;
```

```
;HERE WE TURN ON THE PIA AND GO INTO A DELAY ROUTINE TO MAKE THE
;SOUND... AFTER THE DELAY WE SHUT OFF THE PIA FOR A BRIEF TIME TO
;SEPERATE THE CODE LETTERS
    6059    A9F7      SEND    LDA    #$F7          ;LOAD OUTPUT MASK
    605B    8D00D3            STA     SIGIN        ;TURN PIA ON
    605E    20EC06            JSR     JSRDLY       ;GOTO DELAY
    6061    A9FF              LDA    #$FF          ;LOAD OUTPUT MASK
    6063    8D00D3            STA     SIGIN        ;TURN PIA OFF
    6066    A001              LDY    #$01          ;LOAD GAP
    6068    20EC06            JSR     JSRDLY       ;GOTO DELAY
    606B    CA                DEX                  ;CHECK COUNT
    606C    10E0              BPL     NEXT         ;RETURN FOR NEXT SIGNAL
    606E    A002      FINI    LDY    #$02          ;LOAD A SPACE
    6070    20EC06            JSR     JSRDLY       ;GOTO DELAY
    6073    A9FF              LDA    #$FF          ;LOAD ATRACT FLAG
    6075    854D              STA    $D4           ;STORE IT
    6077    6CFE06            JMP    $(JMPSTR)     ;INDIRECT JUMP TO START
;
;
;THIS ROUTINE OUTPUTS A SPACE ON THE SCREEN AND DELAYS THE CODE
;OUTPUT FOR A TIME TO SIMULATE A SPACE
    607A    A920      SPACE   LDA    #$20          ;LOAD SPACE CHARACTER
    607C    20A4F6            JSR     $F6A4        ;OUTPUT SPACE
    607F    A007              LDY    #$07          ;LOAD SPACE COUNT
    6081    20EC06            JSR     JSRDLY       ;GOTO DELAY
    6084    6CFE06            JMP    $(JMPSTR)     ;INDIRECT JUMP TO START
;
;
;THIS SECTION IS THE DELAY... IT USES THE VALUE STORED IN THE SPEED
;REGISTER AS A TIMER AND THE VALUE OF Y AS THE LENGTH...THE BIGGER
;THE SPEED VALUE, THE SLOWER THE CODE... IT USES THE SOFTWARE
;CLOCK AS THE COUNT-DOWN TIMER
    6087    8A        DELAY   TXA                  ;SAVE THE COUNT
    6088    48                PHA
    6089    A200              LDX    #$00
    608B    A5DE              LDA     SPEED    ;LOAD SPEED
    608D    8D2002    D2      STA     CLOCK        ;STUFF INTO TIMER
    6090    EC2002    D3      CPX     CLOCK        ;WAIT FOR IT TO FINISH
    6093    D0FB              BNE     D3
    6095    88                DEY                  ;CHECK FOR DIT DAH OR SPACE
    6096    D0F3              BNE     D2
    6098    68                PLA                  ;RETREIVE COUNT
    6099    AA                TAX

    609A    60                RTS                  ;AND RETURN
    609B    60        RETURN  RTS
;
;
;THIS SECTION IS USED FOR RANDOM CODE PRACTICE... TO ADDRESS IT
;CHANGE THE VALUE STORED IN JMPSTR TO POINT TO THE BEGINNING OF
;THIS ROUTINE...MAKE SURE THE TEMP REGISTER IS ZEROED... THE
;ROUTINE LOADS A VALUE FROM THE RANDOM GENERATOR AND DIVIDES IT
;UNTIL IT IS WITHIN THE LIMITS OF THE CODE TABLE... THE ROUTINE
;OUTPUTS FIVE CHARACTERS FOLLOWED BY A SPACE... THE ROUTINE EXITS
;BY RESTORING THE PIA TO INPUT
    609C    A927      RANDU   LDA    #$27          ;LOAD LOGO CODE
    609E    CDFC02            CMP     LETT         ;CHECK FOR THAT KEY
    60A1    F01E              BEQ     ATLAST       ;EXIT IF HIT
    60A3    A900              LDA    #$00
    60A5    C5D6              CMP     TEMP         ;CHECK FOR SET END
    60A7    D007              BNE     MORE         ;BRIF NOT DONE
    60A9    A905              LDA    #$05          ;LOAD FOR FIVE CHAR SET
    60AB    85D6              STA     TEMP         ;STORE IN REGISTER
    60AD    18                CLC                  ;CLEVER WAY TO AVOID
    60AE    90CA              BCC     SPACE        ;INDIRECT JUMP
    60B0    C6D6      MORE    DEC     TEMP         ;KEEP COUNTER RUNNING
    60B2    AD0AD2            LDA     RANDOM       ;GET A RANDOM VALUE
    60B5    C92F      SHRINK  CMP    #$2F          ;CHECK ITS OFFSET
    60B7    9004              BCC     GOOD         ;BRIF OFFSET IS IN THE TABLE
    60B9    4A                LSR       A          ;DIVIDE UNTIL ITS GOOD
    60BA    18                CLC                  ;JUMP BACK
    60BB    90F8              BCC     SHRINK       ;AND DIVIDE AGAIN
    60BD    AA        GOOD    TAX                  ;SAVE OFFSET
    60BE    6CEA06            JMP    $(JMPMOR)     ;AND JUMP INDIRECT TO MORSE
    60C1    20DAE6    ATLAST  JSR     $E6DA        ;OS ROUTINE TO INITIALIZE PIA
    60C4    A9FF              LDA    #$FF          ;BLANK OUT THE LOGO KEY
    60C6    8DFC02            STA     LETT
    60C9    60                RTS
```

```
                *=TABST
0640    200E            .BYTE $20,$0E,$22,$26,$32,$1F,$1E,$1A
0642    2226
0644    321F
0646    1E1A
0648    181D            .BYTE $18,$1D,$1B,$33,$35,$30,$42,$02
064A    1B33
064C    3530
064E    4202
0650    360F            .BYTE $36,$0F,$37,$66,$75,$3F,$15,$12
0652    3766
0654    753F
0656    1512
0658    3A2A            .BYTE $3A,$2A,$38,$3D,$39,$0D,$01,$05
065A    383D
065C    390D
065E    0105
0660    0025            .BYTE $00,$25,$23,$08,$0A,$2F,$28,$3E
0662    2308
0664    0A2F
0666    283E
0668    2D0B            .BYTE $2D,$0B,$10,$2E,$16,$2B,$17,$73
066A    102E
066C    162B
066E    1773
0670    3155            .BYTE $31,$55,$32,$3F,$2F,$27,$23,$21
0672    323F
0674    2F27
0676    2321
0678    2030            .BYTE $28,$30,$38,$3C,$3E,$78,$6A,$2A
067A    383C
067C    3E78
067E    6A2A
0680    0045            .BYTE $00,$45,$4C,$28,$05,$18,$1A,$0C
0682    4C28
0684    0518
0686    1A0C
0688    0212            .BYTE $02,$12,$0E,$10,$04,$17,$0D,$14
068A    0E10
068C    0417
068E    0D14
0690    0706            .BYTE $07,$06,$0F,$16,$1D,$0A,$08,$03
0692    0F16
0694    1D0A
0696    0803
0698    0911            .BYTE $09,$11,$0B,$19,$1B,$1C
069A    0B19
069C    1B1C
```



'GATOR' Micropainter file
created by Bruce Masters
printed on an NEC 8023A-C

TERMINET to ATARI

by Adrian Bordelon, KA5BFX

In response to a note from Jack, WD8BNG, I thought I would offer some help on the G.E. Terminet printer. I have a model 120 hooked to my ATARI 800/850 Interface using the parallel port and the operation is VERY fast!

The following info would only be useful if you had the same model printer as I, but if you have a model 300 or 1200, then they should be ready to hook-up via their RS-232C ports (standard). Now, on to the info on the model 120...

The following modification will allow use of the G.E. Terminet Model 120 (usually an RS-232C interface) to be interfaced to the ATARI 850's parallel port. First, remove the SAUX (used as the interface internally to change from serial to parallel data). Then purchase, exchange or otherwise scrounge the same type of connectors as used on the G.E. boards and use them to make the following straps on the mother board where the SAUX board mated:

Signal (ATARI)	Strap	to	DB25 Pin	'850 Connector Pin
Strobe	A25	A26	24	1
D1(D0)	A23	B11	8	2
D2(D1)	A06	A39	13	3
D3(D2)	A10	B13	12	4
D4(D3)	A14	A46	15	5
D5(D4)	A13	B07	20	6
D6(D5)	A03	A51	4	7
D7(D6)	A08	B06	16	8
D8(D7)	A01	A53	18	15
BUSY	A07	A45	14	13
FAULT	A15	B16	11	12
GROUND			7	11

In addition to the above, the following strap settings are necessary on the HINT board in the Terminet printer bussel:

STRAP	SET	REASON
J1	IN	
J2	OUT	
J3	IN	
J19	IN	
J20	OUT	
J23	IN	
J24	OUT	
J11	OUT	
J13	IN	
J12	OUT	--- AUTO LF ON DECODED CR (EOL)
J21	OUT	
J22	IN	
J32	OUT	
J31	IN	
J29	OUT	
J30	IN	
J27	OUT	
J26	IN	
J34	OUT	
J33	IN	

That's all there is to it! I know it looks confusing, but it's not really. Mind you, this applies only to the model 120. I may be able to help if your model is other than this one but my guess is that this is the one you will have. To my knowledge, the models 300 and 1200 cannot be made parallel data feed due to the fact that they use a different type of bussel arrangement and that to many multi-function cards are involved. But, again, they operate standard RS232C serial format.

If you have any questions, please feel free to write and I'll try to help. I may be able to borrow manuals if the need arises. Good luck and maybe I'll see you on the ATARI Micronet some Sunday!

73,

Adrian, KA5BFX

VISUAL INDICATORS
by Tom Heckhaus, SWL

These little circuits will enable you to switch off the speaker in your ATARI 400/800 computers and add a visual indication of keyboard "clicks" and CSAVE/CLOAD prompts. Hearing-impared persons may also find it useful.

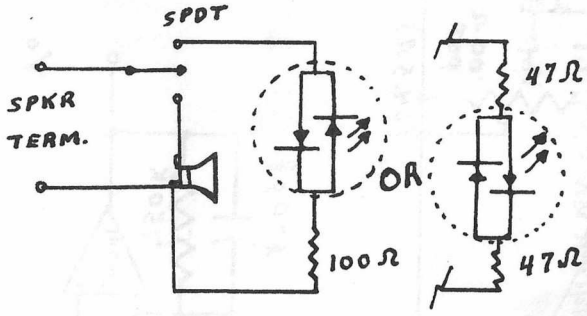
I mounted the SPDT switch to the left of the #1 joystick port on my '400. The chrome LED holder and tri-state LED went to the lower-left of the ROM cart area.

At first I used the LED alone. With a CSAVE command it glowed a very bright yellow. Fearing too much current was being passed, I added a 100 ohm resistor as a current-limiting device (and short-circuit protection). The resistor unbalances the AC a bit and now the LED glows green. (Fig. 1).

* Note: Putting two (2) 47 ohm resistors in series with each leg of the diode should balance the AC and allow it to glow yellow. (Fig. 2). Tom Heckhaus

Parts List

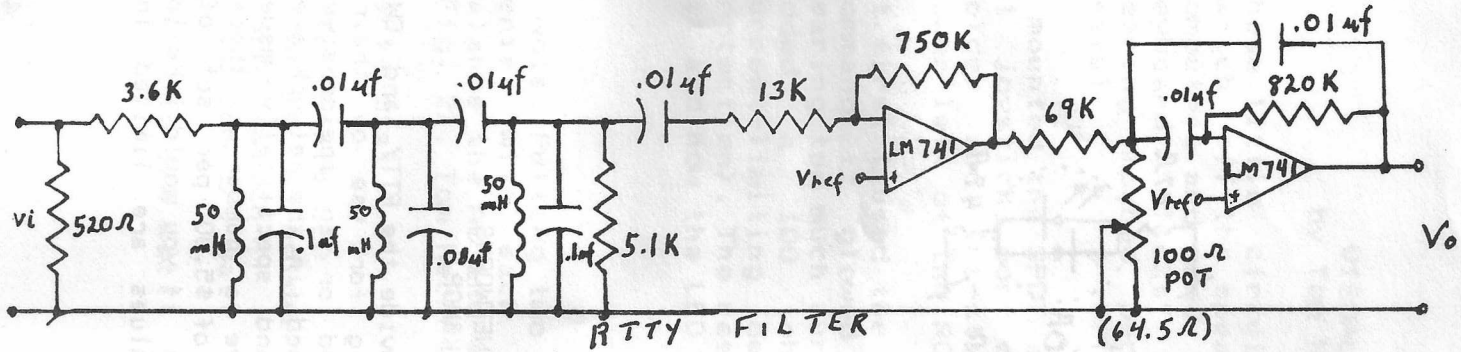
1 Tri-state LED	RS# 276-035
1 Chrome LED-holder	RS# 276-080
1 SPDT switch	RS# 275-625
1 100 ohm 1/4 watt	resistor
or	
2 47 ohm 1/4 watt	resistor

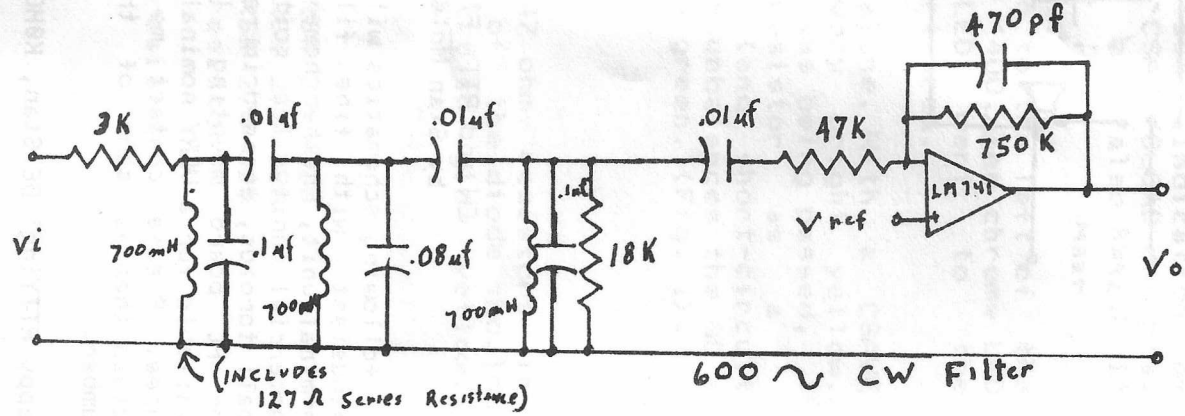


CW and RTTY FILTER NETWORKS
by Stan Molstad, K0HGP

The following schematics will provide the RTTY and CW enthusiast with fine filtering for use on their terminal unit, whether home-brewed or an inexpensive commercial unit. The coils used in the network are small toroids, encased in resin and specifically made for PC board mounting. I have a number of these available for a VERY nominal fee of $5.00 per set of three. Please contact me soon if you would like to obtain them as some of the values are limited in number.

Happy RTTYing! DE Stan, K0HGP


3.6K
.01μf
.01μf
.01μf
13K
750K
69K
.01μf
820K
.01μf
Vi
520Ω
50 mH
.1μf
50 mH
.08μf
50 mH
.1μf
5.1K
LM741
Vref
LM741
Vref
100Ω POT
Vo
RTTY FILTER
(64.5Ω)

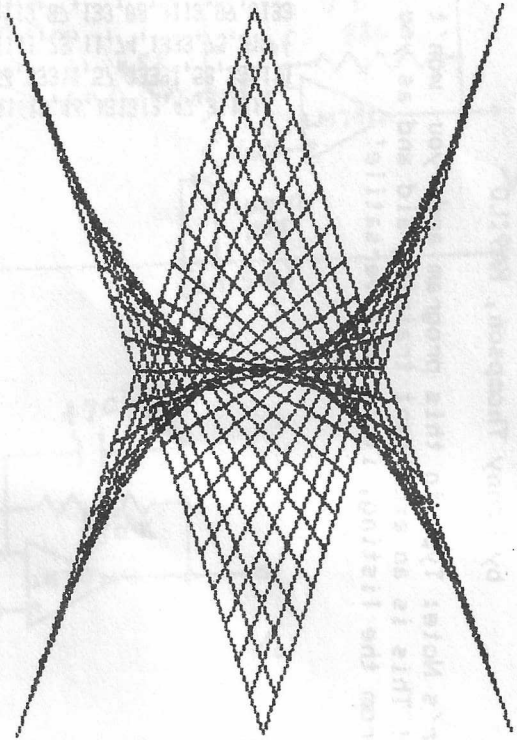

470pf
3K
.01μf
.01μf
.01μf
47K
750K
Vi
700mH
.1μf
700mH
.08μf
700mH
.1μf
18K
LM741
Vref
Vo
(INCLUDES 127Ω Series Resistance)
600 ~ CW Filter

RADIO STATION MORSE
THE UNTIMATE CW TRAINER
by Denny Thompson, KA9ILD

Editor's Note: Type in this program and you won't be sorry! This is an excellent training aid and as you can see from the listing, it is very versatile!

```
10 GRAPHICS 0:POKE 752,1
20 DL=PEEK(560)+PEEK(561)*256
30 DL=DL+4
40 POKE DL+5,7:POKE DL+6,7
50 POKE DL+12,6:POKE DL+13,6
60 POKE DL+28,65:POKE DL+29,PEEK(560)
70 POKE DL+30,PEEK(561):? CHR$(125)
80 SETCOLOR 2,9,0:SETCOLOR 4,9,0
90 POSITION 4,4:? "RADIO STATION          MORSE"
100 POSITION 4,10:? "the ultimate":POSITION 25,10:? "cw trainer"
110 POSITION 8,17:? "  KA9ILD - Denny Thompson"
999 RESTORE
1000 DIM ASC(2),A$(10),MORSE(91,10)
1010 DIM B$(400),C$(100),G$(40)
1020 V=1
1050 READ ASC,A$
1055 MORSE(ASC,1)=LEN(A$)
1060 IF A$="END" THEN GOTO 1300
1065 C$(V,V)=CHR$(ASC):V=V+1
1070 FOR X=2 TO (LEN(A$))+1
1080 MORSE(ASC,X)=VAL(A$(X-1,X-1))
1090 NEXT X
1100 GOTO 1050
1200 DATA 32,111,34,131131,36,1113113,39,133331,40,313313,41,313313,44,331133,45,311113,46,131313,47,31131
1210 DATA 48,33333,49,13333,50,11333,51,11133,52,11113,53,11111,54,31111,55,33111,56,33311,57,33331,58,333111
1220 DATA 59,313131,63,113311,65,13,66,3111,67,3131,68,311,69,1,70,1131,71,331,72,1111,73,11,74,1333,75,313
1230 DATA 76,1311,77,33,78,31,79,333,80,1331,81,3313,82,131,83,111,84,3,85,113,86,1113,87,133,88,3113,89,3133
1240 DATA 90,3311,91,END
1250 POKE 764,255
```

```
1299 REM ***** TITLE PAGE *****
1300 WPM=(1/15)*190:? :B$="PRESS ANY KEY ":DISPLA=2200:GOTO 1510
1400 REM ***** WORDS/  *****
1405 REM ***** MINUTE  *****
1410 GRAPHICS 2:POSITION 1,9:PRINT #6;"  WORDS/MINUTE=";
1420 INPUT WPM:? #6;WPM;:POKE 764,255
1425 FOR DLAY=1 TO 400:NEXT DLAY
1430 WPM=(1/WPM)*190
1450 GOTO DISPLA
1500 GRAPHICS 2:? #6;")":POSITION 5,9:? #6;"INPUT TEXT":INPUT B$:POKE 764,255
1501 FOR DLAY=1 TO 200:NEXT DLAY
1502 GOTO DISPLA
1503 REM ****   SOUND    ****
1504 REM **** SUBROUTINE ****
1505 GRAPHICS 18:? #6;")":SETCOLOR 4,10,6:SETCOLOR 0,2,0:SETCOLOR 2,10,6
1510 FOR Q=1 TO LEN(B$)
1512 IF (PEEK(85)=0)=1 THEN ? #6;" ";
1513 IF (PEEK(85)=19)=1 THEN ? #6;"  ";
1514 T=ASC(B$(Q,Q)):? #6;CHR$(T);
1516 IF T=32 THEN 2050
1520 FOR QQ=1 TO MORSE(T,1)
2000 SOUND 0,40,10,15
2010 FOR W=1 TO MORSE(T,QQ+1)*WPM:NEXT W
2020 SOUND 0,0,0,0
2025 FOR W=1 TO WPM:NEXT W
2030 NEXT QQ
2035 FOR CSPACE=1 TO 3*WPM:NEXT CSPACE
2040 NEXT Q
2045 GOTO 2057
```

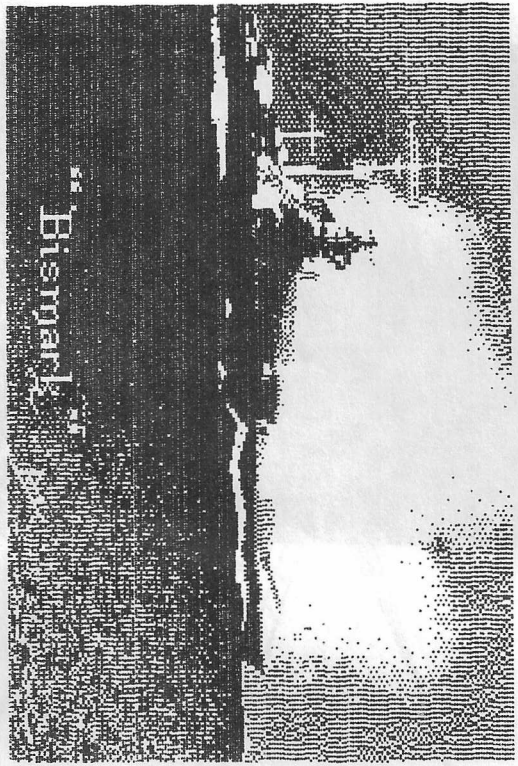


```
2050 FOR WSPACE=1 TO 7*WPM:NEXT WSPACE
2055 TRAP 2200:NEXT Q
2057 SETCOLOR 0,2,4:POKE 764,255
2058 IF PEEK(764)=255 THEN 2058
2059 GRAPHICS 18:GOTO DISPLA
2060 REM **** DISPLAY B1 ****
2061 GRAPHICS 18:? #6;"}":POSITION 5,2:? #6;"   PRESS":POSITION 2,4:? #6;"OPTION NEW TEXT"
2062 POSITION 2,6:? #6;"SELECT NEW SPEED":POSITION 2,8:? #6;"START  FOR CODE"
2063 POKE 53279,0:DISPLA=2060:POKE 764,255
2064 FOR DLAY=1 TO 200:NEXT DLAY
2065 IF PEEK(53279)=5 THEN GOTO 1410
2070 IF PEEK(53279)=6 THEN GOTO 1505
2075 IF PEEK(53279)=3 THEN GOTO 1500
2085 IF PEEK(764)=12 THEN GOTO 2300
2095 GOTO 2065
2098 REM ***** MAIN ****
2099 REM ***** MENU ****
2200 GRAPHICS 18
2210 ? #6;"}":POSITION 6,1:? #6;"MAIN MENU"
2220 POSITION 2,4:? #6;"OPTION TO SEND"
2225 POKE 53279,0:POKE 764,255:DISPLA=2200
2230 POSITION 2,6:? #6;"SELECT TO RECEIVE"
2235 POSITION 2,8:? #6;"START  WORDS/MIN"
2240 FOR DLAY=1 TO 200:NEXT DLAY
2250 IF PEEK(53279)=5 THEN 2300
2260 IF PEEK(53279)=3 THEN 3000
2270 IF PEEK(53279)=6 THEN 1405
2280 GOTO 2250
2299 REM **** DISPLAY B ****
```

'FRANKIE' Micropainter file
created by Bruce Masters
printed on an NEC 8023A-C

```
2300 ? #6;"}":POSITION 2,2:? #6;"PRESS TO RECEIVE"
2310 POSITION 2,4:? #6;"OPTION OWN TEXT"
2320 POSITION 2,6:? #6;"SELECT RANDOM TEXT"
2325 POKE 53279,0:POKE 764,255
2326 FOR DLAY=1 TO 200:NEXT DLAY
2330 IF PEEK(53279)=5 THEN 2400
2340 IF PEEK(53279)=3 THEN 2060
2345 IF PEEK(764)=12 THEN 2200
2350 GOTO 2330
2399 REM XXXX DISPLAY B2 XXXX
2400 ? #6;"}":POSITION 5,2:? #6;"RANDOM TYPE"
2410 POSITION 2,4:? #6;"OPTION ALPHA-"
2415 POSITION 9,5:? #6;"NUMERICS"
2420 POSITION 2,6:? #6;"SELECT GROUPS"
2425 POSITION 2,8:? #6;"START  OWN"
2426 FOR DLAY=1 TO 200:NEXT DLAY
2427 POKE 53279,0:POKE 764,255:DISPLA=2400
2430 IF PEEK(53279)=5 THEN 3200
2440 IF PEEK(53279)=3 THEN 3305
2445 IF PEEK(53279)=6 THEN 2500
2446 IF PEEK(764)=12 THEN 2300
2450 GOTO 2430
2500 REM XXXX DISPLAY B2A XXXX
2510 GRAPHICS 2:? #6;"}    RANDOM INPUT":INPUT G$
2520 FOR DLAY=1 TO 200:NEXT DLAY
2530 DISPLA=2400:GRAPHICS 18:GOTO 3310
2999 REM XXXX DISPLAY A XXXX
3000 GRAPHICS 17:SETCOLOR 4,3,4:SETCOLOR 0,10,10:? #6;"   INTERNATIONAL":? #6;"   MORSE CODE"
3001 ? #6;""
```

'BISMARK' Micropainter file
created by Bruce Masters
printed on an NEC 8023A-C



May 16, 1983

Dear Al,

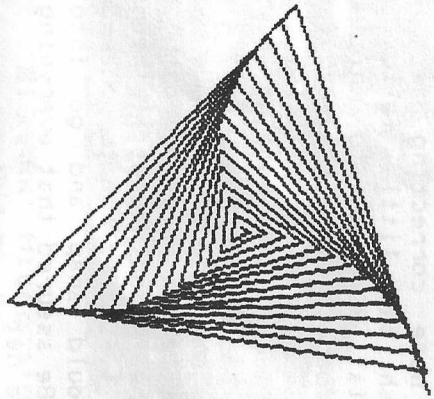
Thank you for your note correcting my 800's mistake! I did admonish the little fellow and threatened him with a frontal ROMotomy to which he replied, "Garbage in - garbage out... it's your fault carbon-unit. My memory, 'though volatile, is infallible! Just make sure you brain is keeping up with your fingers when you tell me something!". I told him he could be replaced with a Commodore 64 and he just sat there producing little chuckles through his keyboard speaker! The more he thought about it the more he convulsed with laughter. I finally had to power-down for fear that his OS would snap and go into a zombie-like (Z-80) state. Be assured that everyhing is correced, Al! See you on the net!

73,

Jack, WD8BNG

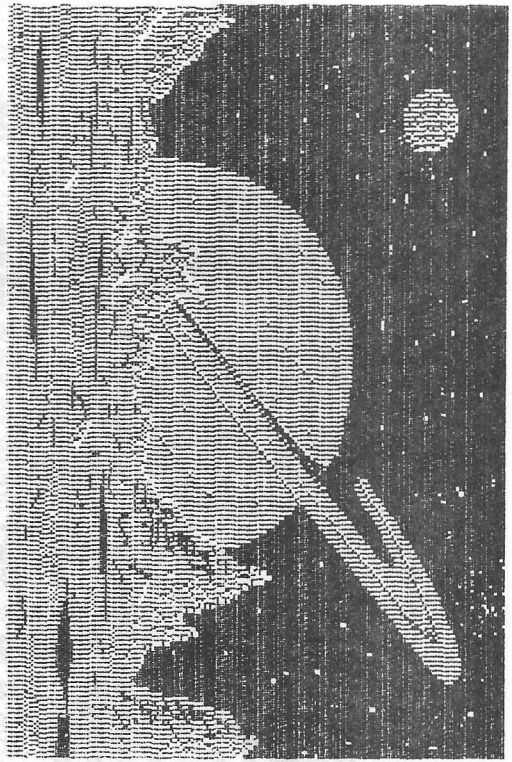
```
3005 ? #6;" A%-  I%%   0--%-"
3006 ? #6;" B-%%% J%--- R%-%"
3007 ? #6;" C-%-% K-%-  S%%%"
3008 ? #6;" D-%% L%-%% T-"
3009 ? #6;" E%   M--   U%%-"
3010 ? #6;" F%%-% N-%  V%%%-"
3011 ? #6;" G--% O---  W%--"
3012 ? #6;" H%%%% P%--% X-%%-"
3014 ? #6;"    Y-%-- Z--%%"
3015 ? #6;""
3016 ? #6;" 1 %----  ? %%--%%"
3017 ? #6;" 2 %%---  , --%%--"
3018 ? #6;" 3 %%%--  . %-%-%-"
3019 ? #6;" 4 %%%%- / -%%-%"
3020 ? #6;" 5 %%%%% []-%--%-"
3021 ? #6;" 6 -%%%% : ---%%%"
3022 ? #6;" 7 ---%%% ; -%-%-%"
3023 ? #6;" 8 ----%% < %-----%"
3025 ? #6;" 9 -----% $ %%%-%%-"
3026 ? #6;" 0 ------"
3027 ? #6;" RETURN TO EXIT"
3098 REM %%%% SOUND ROUTINE %%%%
3099 POKE 764,255
3100 IF STRIG(0)=0 THEN SOUND 0,40,10,10
3110 IF STRIG(0)=1 THEN SOUND 0,0,10,1
3115 IF STICK(0)=14 THEN 5000
3120 IF PEEK(764)=12 THEN 2200
3125 GOTO 3100
3199 REM %%%% DISPLAY B2C %%%%
```

```
3200 POKE 53279,0:POKE 764,255
3210 GRAPHICS 18:DISPLA=3200
3215 ? #6;"}":POSITION 4,2
3220 ? #6;"CODE GROUPS":POSITION 2,4
3230 ? #6;"OPTION GROUP #1":POSITION 2,6
3240 ? #6;"SELECT GROUP #2":POSITION 2,8
3245 ? #6;"START  GROUP #3"
3250 FOR DLAY=1 TO 200:NEXT DLAY
3255 B$="  ":DISPLA=3210
3260 G1=1:IF PEEK(53279)=3 THEN GOTO 3300+G1
3270 G1=2:IF PEEK(53279)=5 THEN GOTO 3300+G1
3280 G1=3:IF PEEK(53279)=6 THEN GOTO 3300+G1
3285 IF PEEK(764)=12 THEN 2400
3290 GOTO 3260
3300 REM XXXX CODE GROUPS XXXX
3301 G$="EISH5TMO074":GOTO 3310
3302 G$="ANRK1LFUV923":GOTO 3310
3303 G$="BDGXZ6WJPCQY8":GOTO 3310
3305 G$="ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"
3310 ? #6;"}":POSITION 8,2:? #6;"RANDOM":POSITION 6,4:? #6;"CHARACTERS":POSITION 9,6:? #6;"ARE"
3312 POSITION 5,8:? #6;G$
3315 FOR DLAY=1 TO 200:NEXT DLAY
3330 REM XXXXX ========== XXXXX
4000 REM XXXXX RANDOM TEXT XXXXX
4005 REM XXXXX  GENERATOR  XXXXX
4010 FOR RTEXT=1 TO 198
4015 SP=RTEXT/6
4020 IF SP-INT(SP)=0 THEN GOTO 4080
4030 RT=INT(RND(0)*LEN(G$))+1
```



```
4040 B$(RTEXT,RTEXT)=G$(RT,RT)
4060 NEXT RTEXT
4075 GOTO 1505
4080 B$(RTEXT,RTEXT)=" "
4090 GOTO 4060
5000 DISPLA=5000
5001 OPEN #3,4,0,"K:"
5002 GET #3,KEY
5003 B$=""
5004 B$=CHR$(KEY)
5005 IF STICK(0)=13 THEN CLOSE #3:GOTO 3100
5010 FOR Q=1 TO LEN(B$)
5014 T=ASC(B$(Q,Q))
5016 IF T=32 THEN 5050
5020 FOR QQ=1 TO MORSE(T,1)
5030 SOUND 0,40,10,15
5040 FOR W=1 TO MORSE(T,QQ+1)*WPM:NEXT W
5050 SOUND 0,0,0,0
5060 FOR W=1 TO WPM:NEXT W
5070 NEXT QQ
5080 FOR CSPACE=1 TO 3*WPM:NEXT CSPACE
5090 NEXT Q
5100 GOTO 5002
5110 FOR WSPACE=1 TO 7*WPM:NEXT WSPACE
5120 NEXT Q
5140 GOTO 5002
```

'SATURN' Micropainter file
created by Bruce Masters
printed on an NEC 8023A-C



REVIEW by Jim Blain

CIRCUIT LAB

by Mark Davids and Sheldon Leemon, N8SL

"CIRCUIT LAB" is a new instructional program to be released through APX in the Summer catalog. An autobooting program written in ATARI BASIC with machine-language routines, "CIRCUIT LAB" makes extensive use of redefined characters to create what amounts to simple series, parallel, series-parallel and multiple-branch circuits. The joystick is used to place bus lines, switches, resistors, ammeters, light bulbs and volt meters at various places around the desired circuit. Player-Missile Graphics are used to light the bulbs at the appropriate times and a occasionally I had to start tracing the circuit to find out why my layout wasn't working! (My fault, not the program's!)

Execution of the program's visuals is SUPERB and everything seems to work just like a high school breadboard! Perhaps the best application of this program would be in a high school physics class, general science class or novice-general amateur radio class. Do not mistake this as a comprehensive course in DC electronics... there are no amplifiers or biasing circuits that can be developed. However, as a method of teaching the flow of eletrons in DC circuits and the interactions of various components' resistive values, it would be difficult to beat in classroom conditions.

The documentation supplied seems to be quite informative and could be considered a teacher's guide for the lessons to be learned. A few screen dumps are supplied to assist in setting-up and becoming familiar with the program. It is quite evident that the program and documentation are written by a professional instructor. Error handling is relatively good but it is possible to get occasional glitches and these I haven't been able to figure out. Once in a while I will specify that the resistors should fall within a specific range and no matter what I do, the values end up as zero ohms! I am not certain why this happens, but it may be due to calling up certain voltage-resistance range combinations. I will report later on the reasons, if I discover them.

In all, I believe that Mark and Sheldon have done a super job of making LEARNING simple DC circuits easy and fun. I understand that this program won 2nd place in the APX Education Catagory contest. Just a few minutes on the keyboard will convince you that computer-assisted education is the only way to fly!

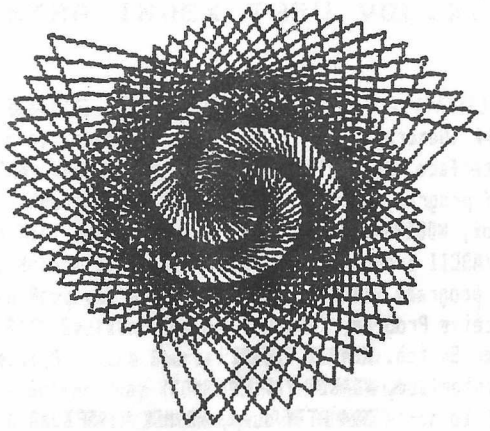
"CIRCUIT LAB" is available from APX or directly from:

Mark Davids

21825 O'Connor

St. Clair Shores, MI 48080

Price: $15



"Ad Astra..." Index
by Randy Agee, WB4BZX

Editor's Note: We are indebted to Randy for his unselfish desire to assist all net members! His latest effort is a compilation of all articles from previous issues of "Ad Astra...". Those of our members who do not have issues mentioned in this compendium may obtain specific information by dropping an SASE to Net HQ. While I may not be able to photocopy every article that is requested, perhaps a short explaination of the jist of the article would suffice. Be kind! DE Jack, WD8BNG

P.S. Anyone want to volunteer to be the historian for the net???

AD ASTRA INDEX THRU VOL.2, #1.

OPERATING SYSTEMS

CITA or GITA V1#1P3
Cure for OS Lockup, ATARI V1#2P19
Rev. "A" vs. "B", WB0QPP V1#3P5
ROM "B" info, Blain V1#6P39

HARDWARE MODS

8K to 16K boards V1#1P6
48/64K Upgrade for 400, MACE V1#4P14
BASIC on/off switch, WB4BZX V1#5P13
Video RFI Fix, WA3WOM V1#5P17
Two Speed 810, WD8MSJ V1#6P31

RTTY/CW (General)

KD7S CW Interface, KD7S V1#2P11
WB0QPP RTTY program, WB0QPP V1#3P16
RTTY Mailbox, WD8BNG V1#4P6
K7JZD RTTY/ASCII programs V1#4PII
K2GTE RTTY programs V1#4PII
KD7S CW Receive Program, KD7S V1#4P23
Dual Printer Switch, WB4BZX V1#4P29
DT-600 TU interface, WD4BIT V1#4P31
NEC 8023A-C to ports 3&4 RTTY dump, WB9MBK V1#5P7
ST-6/DT600 for 110 baud, Blain V1#5P19
Flesher TU-170 and the 400/800, WB9ZDV V1#6P6
TU-170 and RS-232 via 850, KB0UY V1#6P9
Macrotronics "Terminall", WD8BNG V1#6P20
Cheap RTTY Demodulator, WD4BIT V2#1P36

HAMSOFT-KANTRONICS & "THE INTERFACE"

Kantronics (DAYTON!!) review V1#1P9
More on Kantronics Interface, WD8BNG V1#2P5
Hamsoft for the Atari, WD8BNG V1#2P5
Hamsoft/printer on ports 3&4, WD8BNG V1#2P7
HAMSOFT and the FSK-500 TU, WB9FNR V1#4P9
HAMSOFT to MX-80 via ports 3 & 4, WB9FNR V1#4P9
Speaker level reduction with "The Interface", KG2L V1#4P32
TU-170 and "HAMSOFT", KB0UY V1#6P9

W5UGQ/WD4HPL RTTY CW INTERFACE

RTTY/CW Interface, WD4HPL & W5UGQ V1#3P7 & V2#1P5
WD4HPH/W5UGQ construction review, KD8Z V1#4P35
Observations and Mods to Interface, WA5BDU/0 V1#5P12
Use with the ICOM 720, WD4FYB V1#5P24
Modifications to, W3WGN V2#1P12

410 CASSETTE RECORDERS

Operating hint, WB4BZX V1#3P4
Counters & Belt slip, WB9FNR V1#4P10
Curing Cassette Blues, WA2NSM V1#5P9
More on Belt Slip, KC7DG V1#6P16

AD ASTRA INDEX THRU VOL.2, #1.

SOFTWARE REVIEWS AND ANNOUNCMENTS

SOFTWARE PROGRAM LISTINGS

PRINTERS (General)

Model 800 (General)

PERCOM DISK DRIVES

VOICE SYNTHESIZERS

SUPPORT HARDWARE ARTICLES

CONSTRUCTION ARTICLES

400 KEYBOARDS

AD ASTRA INDEX THRU VOL.2, #1.

MEMBER HINTS

MEMBERS WHO SELL ATARI

810 DISK DRIVES

1200XL (General)

LIBARY LISTINGS

REVIEW
"THE CHIP" from Spartan Software
by Gary Miller, W4FCL

"The Chip" is a modification kit for the ATARI 810 disk drive unit that enhances the '810's capabilities tremendously.

Installation instructions provided with "The Chip" are quite clear, but mechanical disassembly/assembly skills are required to effect the following: cut 3 traces on the side-board, install 3 jumper wires and install a new expanded EPROM containing a new instruction set.

An '810 with "The Chip" appears to be a normal drive unit until it is **OPENED**". To "**OPEN**" it you merely insert the disk program "**ARCHIVER-EDITOR**", which is supplied with "The Chip", **THEN** turn the drive on. (This doesn't seem right, but it does no harm.). For the faint-of-heart, the drive may be "OPENED" by the use of keyboard commands after the normal boot-up procedure. "ARCHIVER" will make a backup diskette of any known disk-based program for the ATARI Computer system. Effectively, the good old "smart" 810 has had it's IQ raised to the level that it can now be known as a "super-intellegent" 810! All of those "strange tricks" that have kept you from making back-ups of those delecate diskettes are now obsolete!

In addition, in it's archiver mode, the drive reads the disk on-the-fly, a track at a time. Previously, backing-up a disk with many "bad-sectors" could take up to 2-hours or more before... "The Chip" reads the sector errors as fast or faster than data sectors. Also, sectors without actual data take up very little memory so many disks will copy with only one pass for reading and one pass for writing. The back-up will be an exact duplicate of the original and will function in the same way.

Will it work on everything? So far, for me, it works automatically on all but two of the many hundreds of programs that I have tried. Those two programs were copyable but required some additional instructions from the human (sorry, you can't get out of <u>all</u> the work!)

and then copied fine.

For afficionados of disk-protection-schemes, it handles (automatically) sector errors, seven types of data errors, data errors with data return, reverse tracking for timing routines, multiple sectoring for timeing and/or data return, and additional sectoring (such as 19 intead of 18 sectors on one track). This list is by no means exhaustive, but will give you an idea of it's versatility.

"THE EDITOR" part of the program allows you to construct your own custom "strange format" to include up to 24 sectors/half-sectors on a track.

I have examined the California product that cost $250-$500 (depending on options chosen) and as far as I can determine, this mod does the same job at less cost.

The retail price of this mod was wrongly reported in a previous issue of "Ad Astra..." as $75. The correct retail is $100. For members who want this package, I'm offering it as an introductory special at $85.

DE Gary Miller, W4FCL
D&G Computronics
4505 Shawnee Rd.
Martinez, GA 30907
(404) 860-3700

Editor's Note: I had the pleasure of meeting Phil Seifert of Spartan Software at the Summer CES. Phil is an amiable fellow who really knows the system and the methods of protecting disk software. 'Though I have not experienced the Spartan Software modification firsthand, member John Benkhe reports that it works as advertised and combined with the information in Gary's article, I believe that this is a good product and an especially good value compared to other mods on the market. Jack, WD8BNG

OHNO!
ANOTHER 810 MOD
by Randy T. Agee, WB4BZX

Anyone who has been reading Ad Astra... since it's introduction most likely has come to realize that I am a hardware hacker and am not content unless my screwdriver is handy and soldering iron hot.

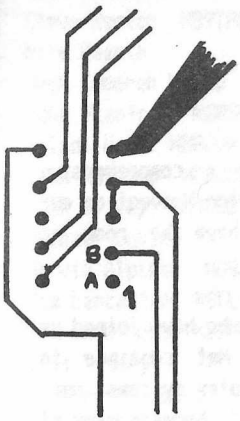
Such was the case several weeks back when I became fed up with punching the left side of diskettes to run dual sided or having to remove and replace the write protect label when I wanted to erase or protect a diskette.

Down came the 810 service manual to check out the protection circuit and in less than 30 minutes the mods were made and the system running again.

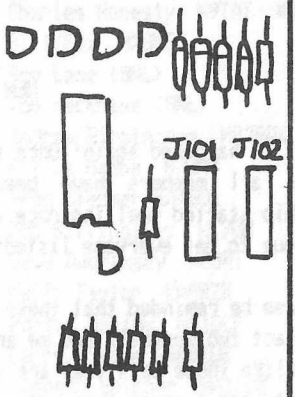
Basically, this is what we have done. If you look at the schematic in figure 3 you will see that by bypassing the phototransistor with a switch we can write to a diskette even if it has the label on it or is not punched. I wired the switch as in figure 4 so a LED was on when we were in the overwrite mode. We can carry this even further by using a DPDT switch with a center off position. By cutting the wire at the top of J101 and attaching it to the switch position indicated by C we have overwrite with the switch down, full protection against overwrite in center position and normal with the switch up. Figure 1 is the back rear of the side PC board on the 810 showing where to tack solder your leads to the switch for the overwrite. Figure 2 is the front side. Whichever mod you choose, if any, is up to you. Mini switches are available from any Radio Shack.

Where and how you mount the extra hardware is up to you, but I suggest you put a DB9, like on your joysticks, in the back of the drive and mount all this in a mini box next to your drive. This will also allow you to make other mods like the one by KC8EL in Vol. 1 #6 of Ad Astra without defacing your 810.

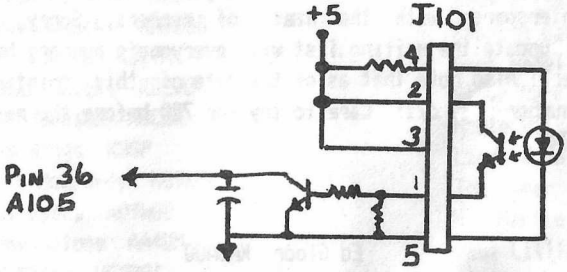
73 Randy WB4BZX


B
A
1

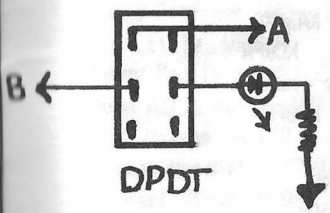
①


J101
J102
D

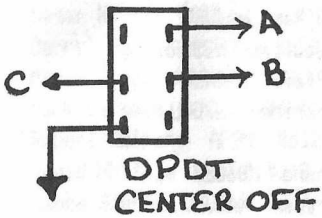
②


+5
J101
4
2
3
1
5
PIN 36
A105

③


A
B
DPDT

④


A
C
B
DPDT
CENTER OFF

⑤

NEW MEMBERS!!!

See! It happened again! Once again my plans to print a comprehensive list of all members have been dashed! After the Dayton Hamvention our membership started swelling once again and we just don't have the room in this issue to get everyone listed!

Please be reminded that these are NEW members (those who have joined us in the last two months!) and we are pleased to see the Net continue to grow! Unlike those nets that are supportive of other computer systems, ours does not see a great turnover in membership. Perhaps it's because many of the "other" systems soon find themselves adorning closet shelves rather than the family room or radio shack!

We welcome all of you to the net and hope to hear from everyone on the air if possible! One thing you will notice is that there are a few "membership numbers" interspersed with the names of members. Sorry, I haven't had time to update the mailing list with everyone's number, but I'll eventually get there! Also note that as of the date of this printing the highest member number is 621! Care to try for 700 before the next issue??? 73, Jack, WD8BNG

Bob Turner N8EAA
Ernst Schuetz KA9JAS
George Hatch W9VMG
John Carter KD4NF
J.P. Keller
Richard Meates N4DTV
Joe Buchanan KA4NCG
Ron Adams KA1WR
Larry Fletcher (SWL)
Bob Menton KG3J
Paul G. O'Ram WA6VEV
Roland Beaulieu WB3CRW
Raymond Pfaff KA4HLG
Jerry Harkrider N7DRU
Dan Lane
Leo Guichard WB6CEJ
David Shrader WA4VKV
Dr. Leo Scanlon M.D. KF5V
Guy Clark WB0NNK
Philip Altman KA6LDA
Mike Caliendo N6BBC
Dave Faucher WA1UQC
Chet Gorski W1PE

Ed Gloor KA5AQQ
Kenneth W. Adams WD9EZG
Jack Katona N8ACQ
Gil Frederick VE4AG
Hal Messer K3ATO
Vic Keller N9GK
Bill Zandrew K9VPS
Al Smochko WB3JEX
Lee Humphrey WD5BTU
Mary Terrill
Sean Terrill KA3KXC
Irvin Koelling KC0FR
Rick Alexander WD5FBW
Robert Huber KB8FC
Mike Braun
Sy Botan K6PWP
Eric Waldemarson
James Snyder WD8NMT
Ronald Fulton
H.S. Gawronski KB2BB
Robert Hunter WB8COW
Stan Mason N8ADV
James Marsh W9NJE

Steve Rector WB9TPU
Mike Dearth
Hugh Cameron N9BLB
John Stanford WB8SVN
Wilson Hall WB8OVR
John Wallace WD8OOS
John C. Day KA4CUB
Michael Squire
Marvin Alpiner WD8JUH
Jim Lancaster K8FL
Clarence Moss K8VPV
Jeff Tussey
Gregory Drezdzon WD9FTZ
Cork Poling W8DHG
Don Hazlett
David Arnold W8DXX
Jack Stoner WA3FTS
James Forkin WA3TFS
Norman Denslow WB3JEL
Steve Nowacki KA9LQG
Gary Ronzio K3YYM
Rodman Bright K8EVD
Bill Johnson N3CGS
Jack Biggs K3SP
Terry McCrarey WB9KQF
Paul Rybka WD9HGO
Danny Malone AA4DM
Ted Ellis VE3BOI
Bob Schenck W8AO
Luis M. De La Vega KP4WI
Harold G. Card Jr. WA4JNG
Mike Holder KA8JKC
Robert Hinkelman
Gerald Kotkowski
Howie De Felice AB2S
Don Montgomery W8PLQ
James Fletcher WD4EFS
Tom Little VE3LJT
Terry Koelling WB4LEJ
John Dalla Valle WB9ZMG #558
Jack Mayson N7ETG
Joseph Marshall K2AU
Michael Halse N9DHF
Bob Uhrik WA2BSP
Bob Briley N4EGE
Ron Whiteside WA4GPD
Robert N. Crowley WB4LRH/I3
Dick Critchell KY2Y #567
R.Evans #568
Paul Wilson #569
Fred Weaver WB8MYO #570
Charles Honesty W9TAT #571
Bill Fox WA3KEU
Roy Lane (SWL)
Tom Heckhaus (SWL)
Andrew Birmingham WB2RQX
Kit La Manna N2BNY
Fred Clemmer KA4GDL
Bob Billingsley KC9UR
Dave Dubrawsky KB3RT
Keith Taylor KA8RZK
Portus Barlow WA1DCP
Larry D'Anna WA3KOK
Kenneth Belcher WA1ZUG
Derek Brown N6CUK
Al Lockhart KA5OSL
Ronald Curtis W8LZW
Richard Meyer K2PBW
James Moore WA3NCM
Dave Erisman WD8PBF
Kenneth Davidson
Mark Nadel WD2AFA
Steve Rechter NA6G
Noble Wilson NC8S
Lance Johnson K1MET
Tom Speer KC6J
Mel Martin VE2DC June 84
Paul Littlejohn KE4DY
ACAOC
William Feist III WB8BZH
John Nastasi Sr. N5CBB
Ben Bunker #601
John Louden WD8NNN #603
Bob Burt KA7A #604
Richard Sutter #605
James Henderson HP1XXY #606
Irvine Green ZS6BPE #607
Cesar N. Mac #608
Darryl Ogden KA2NYV #609
Maria Ogden KA2OBZ #610
Mark Carsman N2CHJ #611
Robert Swirsky AF2M #612
David Whittle N4FND #613
Lance Schultz AI9U #614
Fred Brandeberry WA8KCW #615
Egon Beck DJ1BH # 616 June 85
George Hewitt Sr. N3CCH # 617
Dan Schnarre N9DBB # 618
Calvin Rapaport # 619
Tony Lopez WB5YQT # 620
Dennis Caverly WB8QWL # 621

THE ATARI MICROCOMPUTER NETWORK
JACK McKIRGAN II WD8BNG
4749 S.R. 207 N.E.
Washington C. H., Ohio 43160

BULK RATE
U.S. POSTAGE
PAID
Washington C.H.Ohio
Permit No. 19

Fred Brandeberry WA8KCW #615
Bldg. 360
Argonne IL 60439